IN THE UNITED STATES DISTRICT COURT
WESTERN DISTRICT OF TEXAS
AUSTIN DIVISION

Intellectual Ventures I LLC and **Intellectual Ventures II LLC,**	
Plaintiffs/Counter-Defendants,	**Civil Action No. 1:19-CV-01075-ADA**
v.	
	JURY TRIAL DEMANDED
VMware, Inc.,	
Defendant/Counter-Plaintiff.	

DEFENDANT VMWARE, INC.'S OPENING CLAIM CONSTRUCTION BRIEF

TABLE OF CONTENTS

TABLE OF AUTHORITIES

Page(s)

Cases

Statutes

TABLE OF ABBREVIATIONS

Abbreviation	Full Name
PTO	United States Patent and Trademark Office
POSITA	Person of Ordinary Skill in the Art
Snoeren Decl.	Declaration of Alex Snoeren, Ph.D. Regarding Claim Construction
’686 patent	U.S. Patent No. RE 44,686
’726 patent	U.S. Patent No. RE 42,726
’937 patent	U.S. Patent No. 6,985,937
’937 FH	File History of U.S. Patent No. 6,985,937
’752 patent	U.S. Patent No. 7,949,752
’051 patent	U.S. Patent No. RE 43,051
’818 patent	U.S. Patent No. RE 44,818

TABLE OF EXHIBITS

Exhibit	Title
Ex. 1	Parties' Agreed Constructions to Claim Terms
Ex. 2	Declaration of Alex Snoeren, Ph.D. Regarding Claim Construction
Ex. 3	'937 Patent File History, Applicant Arguments dated November 17, 2003
Ex. 4	U.S. Patent No. 5,933,603 to Vahalia et al.
Ex. 5	'937 Patent File History, Non-Final Office Action dated May 10, 2004
Ex. 6	USPTO Patent Search of claim term "dynamic virtual server mover"
Ex. 7	Terms in the '726 patent that overlap with disputed claim terms in the '686 patent
Ex. 8	The parties' proposals for the terms in the '051 involving multiple recitations of "physical interface(s)" and "tunnel identifier(s)"
Ex. 9	'752 Patent File History, Response to Office Action dated March 16, 2009
Ex. 10	'752 Patent File History, Office Action dated May 5, 2009
Ex. 11	'752 Patent File History, Office Action dated October 30, 2009
Ex. 12	'752 Patent File History, Response to Office Action dated November 13, 2009
Ex. 13	'752 Patent File History, Office Action dated March 16, 2010
Ex. 14	'752 Patent File History, Response to Office Action dated August 20, 2010
Ex. 15	'752 Patent File History, Response to Office Action dated August 5, 2009
Ex. 16	'752 Patent File History, Final Office Action dated November 8, 2010
Ex. 17	'752 Patent File History, Response to Final Office Action dated January 4, 2011
Ex. 18	Report and Recommendation of United States Magistrate Judge, Intellectual Ventures v. HCC Ins. Holdings, Inc., Case No. 6:15-cv-660 (E.D. Tex. August 26, 2016) ("'752 Patent Magistrate Report")
Ex. 19	Charles Aulds, Linux Apache Web Server Administration, 39 (2001) ("Aulds")
Ex. 20	Barry Nusbaum, WebSphere Application Servers: Standard and Advanced Features 45 (1999)
Ex. 21	Ludmila Cherkasova, FLEX: Design and Management Strategy for Scalable Web Hosting Service, 14–15 (Oct. 1999)
Ex. 22	October 14, 2003 Amendment & Remarks, U.S. Patent Appl. No. 09/526,980
Ex. 23	Physical Interface, The IEEE Standard Dictionary of Electrical and Electronics Terms (6th ed., 1996)
Ex. 24	RE43,051 (U.S. Patent Appl. No. 11/858,091) Patent File History, May 17, 2010 Office Action
Ex. 25	U.S. Patent No. 6,286,047 ("Ramanathan")
Ex. 26	U.S. Patent No. 6,247,057 ("Barrera")
Ex. 27	Webster definition of consume
Ex. 28	'752 Patent List of Disputed Claim Terms
Ex. 29	BEN LAURIE AND PETER LAURIE, APACHE: THE DEFINITIVE GUIDE 163, 177, 242–43, 295 (1999)
Ex. 30	*Virtual Server*, MICROSOFT COMPUTER DICTIONARY (5th ed. 2002)
Ex. 31	'789 Patent (Orig. Patent) prosecution history, 2009.08.25 Resp. to Office Action at 12
Ex. 32	Competing Parties' Proposals for the '818 Patent
Ex. 33	Grotto Networking, available at https://www.grotto-networking.com/BBQoS.html
Ex. 34	http://tldp.org/HOWTO/Traffic-Control-HOWTO/index.html

Exhibit	**Title**
Ex. 35	https://lartc.org/howto/lartc.qdisc.classful.html#AEN1071
Ex. 36	Linux Advanced Routing & Traffic Control HOWTO, located at http://www.oamk.fi/~jukkao/lartc.pdf
Ex. 37	Traffic Control HOWTO, Version 1.0.2, Martin A. Brown, located at https://www.tldp.org/HOWTO/html_single/Traffic-Control-HOWTO/
Ex. 38	Traffic Control using tcng and HTB HOWTO, Version 1.0.1, Martin A. Brown, April 2006, located at http://linux-ip.net/articles/Traffic-Control-tcng-HTB-HOWTO.html
Ex. 39	Bavier, et al., Operating System Support for Planetary-Scale Network Services, Proceedings of the First Symposium on Networked Systems Design and Implementation (March 2004), located at https://www.usenix.org/legacy/events/nsdi04/tech/full_papers/bavier/bavier.pdf
Ex. 40	Benita, Kernel Korner - Analysis of the HTB Queuing Discipline Software, available at https://www.linuxjournal.com/article/7562 as of January 26, 2005 and printed in the Linux Journal, March 2005.
Ex. 41	U.S. Patent 7,161,904 titled: "System and method for hierarchical metering in a virtual router based network switch" to Hussein et al.
Ex. 42	Bavier et al, Container-based Operating SystemVirtualization: AScalable,High-performance Alternative to Hypervisors, Conference Paper in ACM SIGOPS Operating Systems Review, January 2007, located at http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.1018.1012&rep=rep1&type=pdf
Ex. 43	Valenzuela, J.L., et al., "A Hierarchical Token Bucket Algorithm to Enhance QoS in IEEE 802.11: Proposal, Implementation and Evaluation, IEEE, vol. 4, Sep. 2004 ("Valenzuela Article")
Ex. 44	Email from J. Deblois to M. Rueckheim dated March 3, 2020
Ex. 45	Webster's New Work Telecom Dictionary, Definition of Layer

I. INTRODUCTION

This case is immensely complicated. IV is asserting more than 40 claims, involving diverse technology from four separate patent families, that were developed by three companies acting independently. Indeed, the total number of words in the asserted claims exceeds 3,700, and hundreds of VMware products and features are accused of infringement. This is an unmanageable number of claims and issues for a jury to comprehend in a trial. Despite the significant number of issues, the parties have worked together to agree to constructions for certain terms (see Ex. 1) and have grouped the remaining disputes into the approximately 30 categories discussed herein. These remaining categories can primarily be summarized into five main groups.

First, for many technical terms, such as "hierarchical token bucket resource allocation" and "physical interface," VMware has proposed constructions consistent with the plain and ordinary meaning of the terms and consistent with the intrinsic record disclosure, while IV has simply proposed they be construed as "plain and ordinary meaning" or argued that the terms are "not amenable to construction." For these terms, IV has confirmed during the meet and confer process that it does not intend to offer alternative proposed constructions or interpretations. Thus, under *O2 Micro Int'l Ltd. v. Beyond Innovation Tech. Co.*, 521 F.3d 1351 (Fed. Cir. 2008) the Court should adopt VMware's proposed constructions.

Second, for certain terms, like "modify a resource allocation" and "exhausted," the respective patent applicants significantly narrowed the claim scope during prosecution in response to PTO rejections. For example, the for term "exhausted"—the applicant actually argued that the prior art's art disclosure of the disclaimed scope (reusable resources) was "***the exact opposite***" of the amended claim language. VMware's proposals hold the applicants to their clear disclaimers consistent with the law. IV asks the Court to ignore these clear disclaimers in favor of an improperly broad reading of the claims.

Third, many terms involve Section 112 ¶ 6 means-plus-function considerations. For some of these terms, reciting classic means-for language, the parties agree that Section 112 ¶ 6 applies, but have disputes as to the specific structures. For some terms, there is no supporting structure counseling in favor of indefiniteness. For the others, VMware's proposed structures are properly limited to the specific portions of the specification that disclose the structures—while IV improperly proposes that the Court construe the structures with respect to ambiguous black-box type terminology. For others of these terms, the parties disagree whether Section 112 ¶ 6 applies. VMware's proposals for these terms recognize that the terms' usage of nonce terminology, like a "component," "module," or "program code" for performing a function, requires application of Section 112 ¶ 6 under *Williamson v. Citrix Online, LLC*, 792 F.3d 1339 (Fed. Cir. 2015). IV's proposal, in contrast, would leave these terms broad, despite the limited disclosure of corresponding structures in the specifications.

Fourth, for several disputes, for example the "virtual server" term, IV seeks to improperly change the meaning of technical terms to craft a better infringement read against VMware's products. Finally, for the remaining disputes, such as the various interfaces/tunnels terms in the '051 patent, and the "enforc[e/ing]", "receiv[e/ing]", etc. terms in the '818 patent, VMware's proposals address significant antecedent basis problems, or seek to clarify the complex technical terminology which may be confusing for a jury. IV's proposals reject these essential clarifications without explanation. VMware's proposals should be adopted at least for the reasons set forth herein and in the declaration of its expert Dr. Alex Snoeren submitted herewith.

II. DISPUTED TERMS FROM U.S. PATENT NO. RE44,686 (the "'686 patent")

A. "modif[y/ied] [a] resource allocation" / "modify[ing] [the] computer resources allocated to a virtual server" ('686 patent claims 5-7)

VMware Proposal	IV Proposal

"modif[y/ied] [a] quality of service guarantee" / "modify[ing] [the] quality of service guarantee of a virtual server" *See also* construction of "quality of service guarantee"	"modif[y/ied] set of functions and features of a physical host(s) used in implementing tasks for each virtual server" / "modify[ing] a set of the functions and features of a physical host(s) used in implementing tasks for each virtual server"

VMware's proposal is based on a clear and unmistakable disclaimer by the applicant during the prosecution of U.S. Patent No. 6,985,937 (the "'937 patent"), from which the '686 reissued. VMware's approach to these terms is also consistent with the claims, the specification,[1] and the applicant's own statements regarding the purpose of the claimed invention. By contrast, IV's proposal disregards the intrinsic record. Furthermore, rather than construe the phrase "modifying a *resource allocation*," IV's proposal reads "allocation" entirely out of the claim term and instead seeks to construe the phrase "modifying a resource."

During prosecution of the '937 patent, the examiner issued a rejection based on U.S. Patent No. 5,933,603 ("Vahalia"). Vahalia is generally directed to providing video-on-demand services by dynamically assigning resources to scheduled tasks. Ex. 4 (Vahalia, Abstract). In attempting to overcome the rejection, the applicant argued that "***[a] resource allocation for a process is specified as a quality of service guarantee. Thus, increasing a quality of service guarantee for a process is equivalent to increasing a resource allocation for that process.***" Ex. 3 at 19 (2003-11-17 Response) (emphasis added).[2] The applicant further argued that Vahalia does not anticipate the claims because Vahalia does not disclose increasing a quality of service guarantee. Rather, the applicant argued that "[o]nce Vahalia has scheduled a task, there is no further determination of whether the resource that is handling the task is overloaded. In other words, in Vahalia, the quality

[1] The specifications of the '686, '726 (discussed in Section III), and '937 patents are identical.

[2] *See MBO Laboratories, Inc. v. Becton, Dickinson & Co.*, 474 F.3d 1323, 1327 (Fed. Cir. 2007) (prosecution history of related patents its relevant to the claim construction analysis).

of service of each resource ***is fixed ahead of time and is not altered once a task is scheduled to the resource***." *Id*. (emphasis added).

The PTO relied on the applicant's disavowal regarding this issue, noting that:

> In the claim languages as found in claims 1, 13, 27, 31, 37, 39, and 44, ***Applicant appears to equate 'the quality of [service] guarantee' to 'allocating a portion of resources' and 'increasing the quality of service guarantee' to 'increasing the resources allocated to an associated service.' The slight departure of these phrases from its nominal meanings is being recognized here.*** Although the examiner has adopted Applicant's lexicographer [sic] in the following prior art rejection, it is noted that care must be taken when comparing to its counterpart in the referenced passages (such difference was raised during a recent telephone interview with Applicant's representative).

Ex. 5 (2004-05-10 Office Action) (emphasis added). Thus, in both a response and in a telephonic interview with the examiner, the applicant represented that the terms "resource allocation" and "quality of service guarantee" are equivalent, and the terms "modifying a resource allocation" and "modifying a quality of service guarantee" are equivalent. The examiner unequivocally memorialized this understanding as set forth above, and the applicant did not dispute this interpretation over the subsequent lengthy file history prior to issuance.

IV should be held to this understanding of the claim term. *Digital Biometrics, Inc. v. Identix, Inc.*, 149 F.3d 1335, 1347 (Fed. Cir. 1998) ("The public has a right to rely on such definitive statements made during prosecution. Notice is an important function of the patent prosecution process….").

The claims further support VMware's proposal. For example, claim 1 of the related '726 patent (which is identical to issued claim 1 of the original '937 patent), recites "in response to the virtual server overloaded signal, to modify a resource allocation for the virtual server" and "receive the virtual server resource modification signal and [] determine whether the first physical host is overloaded." In other words, the claims contemplate that after a virtual server is determined to be overloaded, the resource allocation for the virtual server is modified. And *after* the "resource allocation" for the virtual server is already modified, the system may determine that the physical host is overloaded.[3] Such a step only makes sense if modifying the resource allocation of the virtual server refers to modifying the quality of service guarantee of the virtual server, as contemplated by VMware's proposal. Ex. 2 (Snoeren Decl.) ¶ 31. On the other hand, IV's proposal which construes modifying a resource, such as memory or disk space, would involve modifying the physical host in some way. This does not make sense in context of the claims. *Id.*

The specification further supports VMware's proposal. The specification clearly and consistently specifies that a resource allocation refers to a quality of service guarantee, and modifying a resource allocation means modifying a quality of service guarantee. For example:

> ***A resource allocation for a virtual server is specified as a "quality of service guarantee" for that particular server. Each physical host stores quality of service guarantees for the virtual servers it hosts.*** As a physical host performs processes associated with a particular virtual server, the physical host accesses the stored quality of service information to enable the physical host to request the correct quality of service from the operating system kernel of the physical host.

'686 patent, 4:49-56 (emphasis added). Table 1 of the specification, reproduced below, provides another example of the term "resource allocation" referring to a quality of service guarantee.

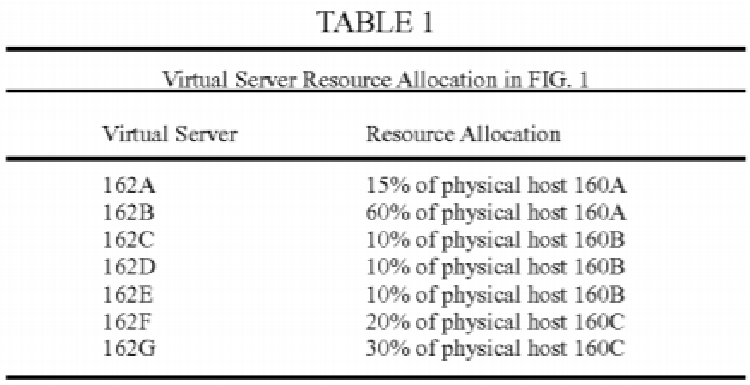


TABLE 1

Virtual Server Resource Allocation in FIG. 1

Virtual Server	Resource Allocation
162A	15% of physical host 160A
162B	60% of physical host 160A
162C	10% of physical host 160B
162D	10% of physical host 160B
162E	10% of physical host 160B
162F	20% of physical host 160C
162G	30% of physical host 160C

[3] The parties agree that "the first physical host is overloaded" should be construed as "the first physical host will not support additional resource allocations at that time."

’686 patent, 4:24-35. As shown in this table, each virtual server is configured with a “resource allocation” defined as a percentage of the resources of a physical host—an arrangement that the specification equates to a quality of service guarantee:

> In one embodiment, ***each individual virtual server 162 has a different quality of service guarantee***. ***Different quality of service guarantees are implemented by allocating different amounts of the resources of each physical host machine 160 to servicing each of the virtual servers 162. Physical host 160 resources may be allocated as percentages of the resources of a particular physical host 160***, or as a particular number of units within a physical host 160 (for example, the operating system may be instructed to allocate X cycles per second to process A and Y cycles per second to process B). In the embodiment shown in FIG. 1, physical host 160 resources are allocated to individual virtual servers 162 as percentages of each physical host 160. ***Table 1 lists the resource allocations of each virtual server 162 as shown in FIG. 1***[.]

’686 patent, 4:9-22 (emphasis added). The specification consistently describes the invention in terms of a quality of service guarantee. *See, e.g.,* ’686 patent, 2:39-42 (“Thus it is desirable to provide a system and method for a virtual server capable of ***providing quality of service guarantees*** for a customer, ***which is also capable of adjusting the quality of service based upon changing customer demand***.”) (emphasis added); *id.*, 3:53-55 (“***The term ‘virtual server’ as used herein refers to a virtual server capable of receiving a quality of service guarantee from a physical host***.”) (emphasis added).

By contrast, for the reasons set forth above, IV’s proposal is not supported by the claims or the specification, and it attempts to disregard an unmistakable prosecution history disclaimer. Moreover, IV’s proposal does not actually construe the term “modify a resource allocation”—instead it appears to only offer a proposed construction of “modify a resource ~~allocation.~~” Specifically, the parties agreed to a construction of the term “resource”[4] and IV has effectively

[4] The parties have agreed that the term resource should be construed as a “set of functions and features of a physical host, such as disk space, memory, network capacity and processing cycles (CPU resources), used in implementing tasks for each virtual server.”

taken this construction and inserted the word "modify" before it for the present term. The fundamental problem with the strategy, however, is that it attempts to read out the actual claim language of "resource allocation." This is improper. *See Pause Tech., LLC v. TiVo, Inc.*, 419 F.3d 1326, 1334 (Fed. Cir. 2005) ("[i]n construing claims, . . . we must give each claim term the respect that it is due"); *Merck & Co. v. Teva Pharm. USA, Inc.*, 395 F.3d 1364, 1372 (Fed. Cir. 2005) ("A claim construction that gives meaning to all the terms of the claim is preferred over one that does not do so."). Resource *allocation* has a specific meaning according to the intrinsic record, and a POSITA would have understood that a *resource* is very different from a *resource allocation*, and that *modifying a resource* is very different from *modifying a resource allocation*. Ex. 2 (Snoeren Decl.) ¶¶ 28-29. For this additional reason, IV's proposal should be rejected.

B. "resource unavailable messages resulting from denied requests to modify a resource allocation" ('686 patent claims 5-7)

VMware Proposal	IV Proposal
"indications that requests by the virtual server for additional resources are either implicitly or explicitly denied, resulting from denied requests to modify a resource allocation" *See also* construction of "modify a resource allocation"	See IV proposals for "resource unavailable messages" and "denied requests to modify a resource allocation" "resource unavailable messages" = "an indication that a request by the virtual server cannot be immediately serviced" "denied requests to modify a resource allocation" = "a request by the virtual server that cannot be immediately serviced"

Both parties have proposed this term for construction. The fundamental dispute regarding this term surrounds IV's attempt to read the phrase "resulting from denied requests to modify a resource allocation" completely out of the proposed term and the claims. As illustrated in the table above, IV's proposal for "resource unavailable messages" is ***nearly identical*** to its proposal for "denied requests to modify a resource allocation." In fact, IV's proposal for the former term renders the latter term obsolete. The Court should reject IV's attempt to remove the language "resulting from denied requests to modify a resource allocation" from the claims and instead adopt VMware's proposed construction which gives due credit to the each and every word in this proposed claim term. *See, e.g., Pause Tech.,* 419 F.3d at 1334; *Merck*, 395 F.3d at 1372.

As to the claim language "resource unavailable messages," VMware and IV's respective proposed constructions are the same as the respective proposed construction for the disputed '726 patent term "resource denial". *See* section III.B. For the reasons set forth below in section III.B, VMware requests that the Court adopt its proposed construction of "an indication that a request by the virtual server for additional resources is either implicitly or explicitly denied" for both terms.

As to the claim language "resulting from denied requests to modify a resource allocation," a proper construction should give meaning to this phrase, rather than ignore it altogether as IV proposes. Specifically, IV's proposals, when read together, would construe "resource unavailable messages resulting from denied requests to modify a resource allocation" as "an indication that a request by the virtual server cannot be immediately serviced [resulting from] *a request by the virtual server that cannot be immediately serviced*." Not only is this proposed construction circular, but it also reads out the claim language reciting that there is a denied request to ***modify* a** resource ***allocation***. A "request by the virtual server that cannot be immediately serviced" (as IV proposes) is clearly not the same thing as denying a "request to modify a resource allocation" (as the claim recites). Ex. 2 (Snoeren Decl.) ¶¶ 32-33. Indeed, IV's proposed construction of this term even ignores its own proposed construction of the term "modify a resource allocation." IV's proposed construction which attempts to remove critical language in the claims should be rejected.

C. "determination that a virtual server is overloaded" ('686 patent claims 5-7)

VMware Proposal	IV Proposal
"determination that an average number of resource denials for a virtual server is beyond a pre-configured threshold"	Plain and ordinary meaning

See also construction of "resource denials"	

This technical term merits construction beyond a "plain and ordinary meaning." The independent claims of the '686 patent (and '726 patent) discuss both virtual server overload and physical host overload. The parties have agreed upon a construction of physical host overload because it is a technical term described in the specification.[5] By this same logic, the Court should construe the technical concept of a virtual server being overloaded as used in the '686 patent.

The concept of overload of a virtual server is described *very differently* in the specification than the concept of overload of a physical host, and the terms therefore should be construed differently. VMware's proposal applies the clear definition of this term from the specification:

> A determination is made 220 as to whether a particular virtual server resource is overloaded. The number of times a particular resource denial is received in a time window is averaged using one of a number of well-known techniques. ***If the average number of denials is beyond a pre-configured threshold, the virtual server is determined 220 to be overloaded for the corresponding resource***. If the virtual server is not determined to be overloaded, the method continues to monitor 210 virtual server resource denials.

'686 patent, 5:42-50 (emphasis added). Adopting any other construction for this term would render the claim invalid for lack of written description. For example, IV's proposed construction of "plain and ordinary meaning" will likely be argued as equivalent to its definition for "virtual server overloaded signal" in the '726 patent as "an indication that a virtual server has been or is being denied resources." But as argued, IV's proposal lacks foundation in the specification, ignores the clear description of this term (reproduced above), and disregards the relationship between virtual server overload and resource denials that is repeatedly set forth in the intrinsic record.

[5] The parties have agreed on a construction of the '686 patent claim term "indication that a first physical host is overloaded" as "indication that a first physical host would not support additional resource allocations at that time" and the '726 patent claim term "the first physical host is overloaded" as "the first physical host will not support additional resource allocations at that time."

IV has refused to explain what it believes the plain and ordinary meaning of this claim term means, or otherwise explain why it disagrees with VMware's proposal. The Court should not construe the term as "plain and ordinary meaning" where, as here, there is a dispute as to the meaning. *O2 Micro*, 521 F.3d at 1351. Therefore, the Court should adopt VMware's proposal.

D. "virtual server" ('686 patent claims 5-7)

VMware Proposal	IV Proposal
"a process executing on a host computer that accepts communication requests, and that is capable of receiving a quality of service guarantee from a physical host"	plain and ordinary meaning; in the alternative: "a virtual machine that resides on a physical server and uses the physical server's resources but has the appearance of being a separate dedicated machine"

The specification defines that: "The term 'virtual server' ***as used herein*** refers to a virtual server capable of receiving a quality of service guarantee from a physical host." '686 patent, 3:53-55. Despite this, IV's proposed construction does not include the language "capable of receiving a quality of service guarantee from a physical host" and should be rejected on that basis alone. The parties also disagree as to the meaning of the term "virtual server" itself. This term is found in the claims of the '686 and '726 patents, which reissued from a patent originally assigned to the company Ensim and filed on May 11, 2000. This term is also found in the claims of the '051 patent, which reissued from a patent originally assigned to Ensim and filed on March 15, 2000. For the reasons set forth below in section V.A, VMware's proposal of the term "virtual server" is necessitated by the intrinsic record and finds additional support in the extrinsic evidence, while IV's proposed construction relies exclusively on extrinsic evidence which it has then modified.

E. "determining that a second physical host can accommodate the requested modified resource allocation" ('686 patent claims 5-7)

VMware Proposal	IV Proposal
Indefinite, or in the alternative: "determining that a second physical host can accommodate the denied requests to modify a resource allocation"	plain and ordinary meaning; in the alternative:

See also construction of "modify a resource allocation"	"determining that a second physical host can accommodate the request(s) by the virtual server that could not be immediately serviced"

This claim term renders the claim 5-7 indefinite. The term includes the phrase "the requested modified resource allocation." However, the claim never introduces "a requested modified resource allocation." *Baldwin Graphic Sys., Inc. v. Siebert, Inc.,* 512 F.3d 1338, 1343 (Fed. Cir. 2008) ("[MPEP] 2173.05(e) describes the need, in most cases, for claim terms to have proper antecedent bases: 'The lack of clarity could arise where a claim refers to "said lever" or "the lever," where the claim contains no earlier recitation or limitation of a lever and where it would be unclear as to what element the limitation was making reference.'"). IV's proposal seeks to construe the term "the requested modified resource allocation" as "the request(s) by the virtual server that could not be immediately serviced," which is nearly identical to its proposals for the terms "resource denials" and "resource unavailable messages." IV's proposed construction has a number of problems. First, it introduces yet another antecedent basis problem by proposing "the request(s)" when the claim language does not recite "a request(s)." Indeed, this apparent change from the singular of the term "the requested modified resource allocation" to IV's plural construction appears to be a concession that the claim term at least lacks antecedent basis as written. Moreover, IV's understanding of this claim term supports the argument that this claim term should, in fact, be construed as indefinite, because it is not possible to assess for the purposes of infringement, which of "the request(s) by the virtual server that could not be immediately serviced" that the second physical is being determined to be able to accommodate. Furthermore, as discussed above, IV's proposal for this term is at odds with its own proposal for the term "modified resource allocation." Although VMware believes this term to be indefinite for the reasons set forth above, it has also proposed an alternative to remedy the antecedent basis problem by referring back to the term "denied requests to modify a resource allocation" which is introduced in the prior claim limitation.

F. "a component configured to receive an indication that a first physical host is overloaded, wherein the indication is based on a determination that a virtual server is overloaded and wherein the determination that a virtual server is overloaded is based on one or more resource unavailable messages resulting from denied requests to modify a resource allocation" ('686 patent claim 7)

VMware Proposal	IV Proposal
Means-plus function term. This term is indefinite. Function: receive an indication that a first physical host is overloaded, wherein the indication is based on a determination that a virtual server is overloaded and wherein the determination that a virtual server is overloaded is based on one or more resource unavailable messages resulting from denied requests to modify a resource allocation Structure: This term is indefinite for a lack of sufficient corresponding structure in the specification.	Not subject to § 112 ¶ 6 - in the alternative: Function: receive an indication that a first physical host is overloaded Structure: Dynamic Resource Configuration Module 100; Physical Hosts 160A-C; Virtual Servers 162A-G

The Court should construe this language which recites "a component configured to … [perform a function]" as subject to 35 U.S.C. § 112 ¶ 6. The term "component" is a nonce term that fails to carry a structural meaning. Ex.2 (Snoeren Decl.) ¶¶ 35-37; *Alarm.com, Inc. v. SecureNet Techs., LLC*, No. CV 15-807-RGA, 2019 WL 3996883, at *6 (D. Del. Aug. 23, 2019) ("While the claim term does not use the words "means," the word "component" is a "nonce" or non-structural word under § 2181 of the Manual of Patent Examining Procedure."); *Amdocs (Israel) Ltd. v. Openet Telecom, Inc.*, No. 110CV910LMBJFA, 2018 WL 1699429, at *19 (E.D. Va. Apr. 6, 2018) ("'component' does not refer to any specifically known structure in the art [… and] is subject to § 112(f)"). Indeed, the term "component" is not found anywhere in the '686 patent aside from claim 7. *See generally*, '686 patent.

Furthermore, claim language reciting what the component is "configured to" do is functional. *MTD Prod. Inc. v. Iancu*, 933 F.3d 1336, 1343 (Fed. Cir. 2019) ("the claim language reciting what the mechanical control assembly is "configured to" do is functional.")

However, the term renders the claim indefinite because the '686 patent does not describe any component that performs the claimed function of "receiv[ing] an indication that a first physical host is overloaded, wherein the indication is based on a determination that a virtual server is overloaded and wherein the determination that a virtual server is overloaded is based on one or more resource unavailable messages resulting from denied requests to modify a resource allocation." There is absolutely no software algorithm disclosed in the original specification for determining that a virtual server is overloaded ***based on one or more resource unavailable messages resulting from denied requests to modify a resource allocation***. *See generally*, '686 patent; Ex. 2 (Snoeren Decl.) ¶¶ 38-39. Furthermore, IV has declined to identify ***any*** portion of the '686 patent that discloses an algorithm for the claimed component. Nor is IV's broad proposal of ***multiple*** components ("Dynamic Resource Configuration Module 100; Physical Hosts 160A-C; Virtual Servers 162A-G") sufficient for providing the requisite structure. *Augme Techs., Inc. v. Yahoo! Inc.*, 755 F.3d 1326, 1338 (Fed. Cir. 2014) ("Simply disclosing a black box that performs the recited function is not a sufficient explanation of the algorithm required to render the means-plus-function term definite.") Since this term is a means-plus-function term and the '686 patent fails to disclose any algorithm to perform the recited function, this term renders claim 7 indefinite.

G. "a component configured to determine that a second physical host can accommodate the requested modified resource allocation" ('686 patent claim 7)

VMware Proposal	IV Proposal
Means-plus function term. This term is indefinite. Function: determine that a second physical host can accommodate the requested modified resource allocation *See also* alternate construction of this term in Section II.C	Not subject to § 112 ¶ 6 - in the alternative: Function: determine that a second physical host can accommodate the requested modified resource allocation

Structure: This term is indefinite for a lack of sufficient corresponding structure in the specification.	Structure: Dynamic Resource Configuration Module 100; Physical Hosts 160A-C; Virtual Servers 162A-G

For the reasons discussed in section II.F, this term is a means-plus-function term subject to 35 U.S.C. § 112 ¶ 6. However, the ’686 patent does not describe any component that performs the claimed function of “determin[ing] that a second physical host can accommodate the requested modified resource allocation.” As discussed in section II.C, to the extent this claim limitation is not determined to be indefinite, the term “the requested modified resource allocation” should be construed to be “the denied requests to modify a resource allocation.” There is no software algorithm, as depicted by a flowchart (or otherwise), that is disclosed in the specification for determining that a second physical host can accommodate the denied requests to modify a resource allocation. *See* ’686 patent; Ex. 2 (Snoeren Decl.) ¶¶ 40-41.

Again, IV has declined to identify portions of the ’686 patent that described an algorithm for the claimed component. As such, this term renders claim 7 invalid as indefinite.

H. “a component configured to generate a physical host transfer signal that indicates a second physical host and to transfer the virtual server from the first physical host to the second physical host if the first physical host is overloaded” (’686 patent claim 7)

VMware Proposal	IV Proposal
Means-plus function term. Function: generate a physical host transfer signal that indicates a second physical host and to transfer the virtual server from the first physical host to the second physical host if the first physical host is overloaded *See also* construction of “virtual server” Structure: Dynamic Virtual Server Mover 140 as described in ’686 Patent, 12:1-28; Figure 6.	Not subject to § 112 ¶ 6 - in the alternative: Function: generate a physical host transfer signal Structure: Dynamic Resource Configuration Module 100; Physical Hosts 160A-C; Virtual Servers 162A-G

For the reasons discussed in Section II.F, this term is a means-plus-function term subject to 35 U.S.C. § 112 ¶ 6. However, the ’686 patent does identify a software algorithm that is associated with performing the claimed function. Figure 6 and the corresponding description at ’686 patent, 12:1-28, describe a component called a “Dynamic Virtual Server Mover” “for transferring a virtual server from one physical host to another physical host.” ’686 patent, 12:1-3.

As shown in the ’686 patent’s Figure 6 and at 12:1-28, the “Dynamic Virtual Server Mover” is disclosed as a specific algorithm that involves storing state information, stopping virtual server processes in the first host, accessing the state information by the second host, starting virtual server processes on the second host, and transferring a user of the virtual server from the first host to the second host. The ’686 patent discloses the mover “may use either ‘make, then break’ timing or ‘break, then make’ timing for the transfer process.” ’686 patent at 12:27-28. Given that this is the only disclosure in the patent—identified by either side—of an algorithm for performing the claimed function, the Court should adopt VMware’s proposal. Ex. 2 (Snoeren Decl.) ¶¶ 42-49.

III. DISPUTED TERMS FROM U.S. PATENT NO. RE42,726 (the “’726 patent”)

A. Terms that overlap with disputed claim terms in the ’686 patent

The terms identified in Ex. 7 overlap with proposed constructions for terms in the ’686 patent discussed above. For the reasons set forth in sections II.A, II.C, and II.D, the Court should adopt VMware’s proposed constructions for these terms.

B. “resource denials” (’726 patent claims 1, 4-5 and 8)

VMware Proposal	IV Proposal
“indications that requests by the virtual server for additional resources are either implicitly or explicitly denied”	“an indication that a request by the virtual server cannot be immediately serviced”

To streamline the dispute, the parties have reached agreement regarding the first part of the construction of “resource denials” – i.e., “indications that requests by the virtual server…” (In contrast with IV, VMware's proposal for this first part of the construction is plural in line with the term resource denial***s*** - i.e., requiring more than one resource denial). The parties, however, disagree on the last part of the construction. VMware's and IV's proposals are drawn from different parts of the specification. VMware's proposal comes from a definitive statement in the specification of "resource denials" – "The virtual server resource monitor 110 monitors different types of resource denials, ***which are*** instances wherein a request for ***additional resources is either implicitly or explicitly denied***." '726 patent, 7:51-54 (emphasis added). By contrast, IV's proposal is drawn from a more tentative description of "resource denials" – "A resource denial ***may refer*** to any request by the virtual server ***that cannot be immediately serviced***…." '726 patent, 2:55-61 (emphasis added). As such, the Court should adopt VMware proposed construction of this term.

C. "quality of service guarantee" ('726 patent claims 1 and 4)

VMware Proposal	IV Proposal
"information that specifies a guaranteed amount of an assigned resource, and that can be dynamically increased/modified"	"a guaranteed resource allotment which can be dynamically increased/modified"

The parties seem to agree that a quality of service guarantee refers to information specifying a guaranteed amount of a resource assignment/allotment that can be dynamically increased/modified. A POSITA would have understood that specifying a "guaranteed" amount of a resource refers to providing a particular amount of an assigned resource that is guaranteed. Ex. 2 (Snoeren Decl.) ¶ 51. This specification of a guaranteed amount can be dynamically increased/modified to specify additional assigned resources if needed. *Id*. The specification explains that conventional static resource assignments would cause a customer who "initially request[ed] a very low level of resources" to "quickly bump up against the limit of the server resources he originally requested" and that therefore "it is desirable to provide a system and method for a virtual server capable of providing quality of service guarantees for a customer, which is also capable of adjusting the quality of service based upon changing customer demand." '726 patent, 2:3-34. The specification further describes quality of service guarantees as follows.

> Different quality of service guarantees are implemented by allocating different amounts of the resources of each physical host machine 160 to servicing each of the virtual servers 162. **Physical host 160 resources may be allocated as percentages of the resources of a particular physical host 160, or as a particular number of units within a physical host 160** [...]. In the embodiment shown in FIG. 1, physical host 160 resources are allocated to individual virtual servers 162 as percentages of each physical host 160. Table 1 lists the resource allocations of each virtual server 162 as shown in FIG. 1:

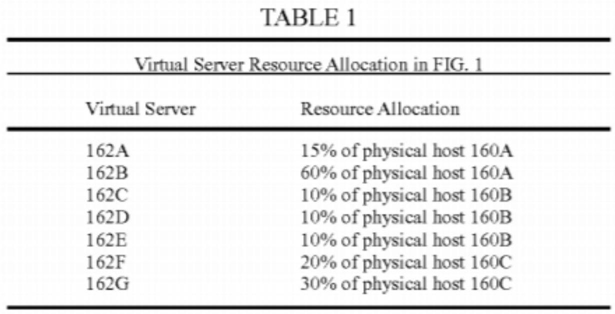


TABLE 1

Virtual Server Resource Allocation in FIG. 1

Virtual Server	Resource Allocation
162A	15% of physical host 160A
162B	60% of physical host 160A
162C	10% of physical host 160B
162D	10% of physical host 160B
162E	10% of physical host 160B
162F	20% of physical host 160C
162G	30% of physical host 160C

'726 patent, 3:67–4:25 (emphasis added).

> As shown in FIG. 1, virtual server 162A consumes 15% of the physical host 160A resources. **This means that 15% of physical host 160A's disk space, memory, network bandwidth, and CPU processing will be dedicated to servicing the needs of virtual server 162A**. [...]
>
> **Each physical host stores quality of service guarantees for the virtual servers it hosts.** As a physical host performs processes associated with a particular virtual server, the physical host accesses the stored quality of service information to enable the physical host to request the correct quality of service from the operating system kernel of the physical host.

'726 patent, 4:32–46 (emphasis added). VMware's proposed construction of this term should be adopted because it is consistent with the specification and provides clarity on the term guaranteed. By contrast, IV's proposal provides no clarity on what is being guaranteed, and thus does not address the apparent dispute between the parties on this term.

D. "a virtual server resource monitor [communicatively coupled to the first physical host and] configured to monitor resource denials and to send a virtual server overloaded signal in response to the resource denials" ('726 patent claims 1 and 5) // "program code for creating a virtual server resource monitor communicatively coupled to the first physical host and configured to monitor resource denials and, in response to the resource denials, to send a virtual server overloaded signal" ('726 patent claim 4)

VMware Proposal	IV Proposal
Means-plus function terms. Function: [creating a virtual server resource monitor communicatively coupled to the first physical host and] monitor resource denials and to send a virtual server overloaded signal in response to the resource denials *See also* construction of the terms "resource denials" and "virtual server overloaded signal" Structure: Virtual Server Resource Monitor 110 as described in '726 Patent, 5:21-65; 7:41-9:46; Figure 3.	Not subject to § 112 ¶ 6 - in the alternative Function monitor resource denials and send a virtual server overloaded signal in response Structure Dynamic Resource Configuration Module 100; Physical Hosts 160A-C

The Court should construe the language in claims 1 and 5 which recites "a virtual server resource monitor … configured to … [perform a function]" as subject to 35 U.S.C. § 112, ¶ 6. The term "virtual server resource monitor" is a black-box term that fails to carry a structural meaning to a POSITA. Ex. 2 (Snoeren Decl.) ¶¶ 53-54; *compare Personal Audio, LLC v. Apple, Inc.*, No. 9:09-cv-111, 2011 WL 11757163, at *21 (E.D. Tex. Jan. 31, 2011) ("If 'computer' or 'processor' is insufficient structure to define the scope of a means-plus-function limitation, the word 'processor' cannot describe sufficient structure when recited directly in a claim limitation itself"). Furthermore, as discussed above, claim language reciting what the component is "configured to" do is functional. *MTD Prod. Inc.*, 933 F.3d at 1343.

Similarly, the Court should construe the language in claim 4 which recites "program code for [performing a function]" as subject to 35 U.S.C. § 112 ¶ 6. The term "program code" is a nonce term that fails to carry a structural meaning. Ex. 2 (Snoeren Decl.) ¶ 55; *Glob. Equity Mgmt.*

(SA) Pty. Ltd. v. Expedia, Inc., 2016 WL 7416132, at *29 (E.D. Tex. Dec. 22, 2016) ("the 'program code for configuring ...' term is governed by § 112, ¶ 6."). Indeed, the term "program code" is not found anywhere in the '726 patent aside from claim 4. *See generally*, '726 patent.

The '726 patent identifies a software algorithm that is associated with performing the claimed function. '726 patent, Figure 3, 5:21-65, 7:41-9:46. Specifically the patent describes a "Virtual Server Resource Monitor" which "monitors different types of resource denials" and "determine[s] if any resources from [a] virtual server [] are overloaded." *Id.* at 7:51-53.

The "Virtual Server Resource Monitor" is disclosed with a specific algorithm that involves monitoring different types of resource denials by selectively intercepting system calls, storing resource denial signals in individual resource denial table(s), determining if the number of resource denials in a time window exceeds a pre-specified threshold to determine if a virtual server is overloaded, and if so, sending a virtual server overloaded signal to the virtual server resource modifier. As this is the only disclosure in the patent of an algorithm for performing the claimed function, the Court should adopt VMware's proposal. Ex. 2 (Snoeren Decl.) ¶¶ 56-61.

E. "a virtual server resource modifier [communicatively coupled to the first physical host and] configured to receive the virtual server overloaded signal and, in response to the virtual server overloaded signal, to modify a resource allocation for the virtual server and to send a virtual server resource modification signal" ('726 patent claims 1 and 5); "program code for creating a virtual server resource modifier communicatively coupled to the first physical host and configured to receive the virtual server overloaded signal and, in response to the virtual server overloaded signal, to modify a resource allocation for the virtual server and to send a virtual server resource modification signal" (claim 4)

VMware Proposal	IV Proposal
Means-plus function terms. Function: [creating a virtual server resource modifier communicatively coupled to the first physical host] receive the virtual server overloaded signal and, in response to the virtual server overloaded signal, to modify a resource allocation for the virtual server and to send a virtual server resource modification signal	Not subject to § 112 ¶ 6 - in the alternative Function receive the virtual server overloaded signal and in response, modify a resource allocation for the virtual server and send a

See also constructions of the terms "virtual server overloaded signal," "modify a resource allocation," and "virtual server" Structure: Virtual Server Resource Modifier 120 as described in '726 Patent, 3:43-48, 3:66-5:4; 5:21-65; 9:47-10:52; Figure 4.	virtual server resource modification signal Structure Dynamic Resource Configuration Module 100; Physical Hosts 160A-C; Virtual Servers 162A-G

The Court should construe this language which recites "a virtual server resource modifier … configured to … [perform a function]" as subject to 35 U.S.C. § 112 ¶ 6. The term "virtual server resource modifier" is a black-box term that fails to carry a structural meaning. Ex. 2 (Snoeren Decl.) ¶¶ 62-63. For the reasons discussed in Section III.D, this term is a means-plus-function term subject to 35 U.S.C. § 112 ¶ 6.

Furthermore, for the reasons discussed in Section III.D, the term "program code for [performing a function]" is a means-plus-function term subject to 35 U.S.C. § 112 ¶ 6. *Id.* at ¶ 64.

The '726 patent identifies a software algorithm that is clearly associated with performing the claimed function. Figure 4, 3:43-48, 3:66-5:4, 5:21-65, and 9:47-10:52 of the '726 patent, describe a component called a "Virtual Server Resource Modifier" which "dynamically modifies the resource allocations of the virtual servers [] on an as-needed basis." '726 patent, 5:7-9.

The "Virtual Server Resource Modifier" is disclosed with a specific algorithm that involves receiving a virtual server overloaded signal from the virtual server resource monitor, determining whether the received signal falls within a pre-specified hysteresis time window, and if not, increasing the virtual server resource allocation (i.e., quality of service guarantee). As this is the only disclosure in the patent—identified by either side—of an algorithm for performing the claimed function, the Court should adopt VMware's proposal. Ex. 2 (Snoeren Decl.) ¶¶ 65-69.

F. "a load balanc[ing/er] [module] [communicatively coupled to the plurality of physical hosts and] configured to receive the virtual server resource modification signal and to determine whether the first physical host is overloaded and, in response to a determination that the first physical host is overloaded, to send a physical host transfer signal that indicates a second physical host" ('726 patent claims 1 and 5) // "program code for creating a load balancing module communicatively coupled to the plurality of physical hosts and configured to receive the virtual server resource modification signal and to determine whether the first physical host is overloaded and, in response to a determination that the first physical host is overloaded, to send a physical host transfer signal that indicates a second physical host" ('726 patent claim 4)

VMware Proposal	IV Proposal
Means-plus function terms. Function: [creating a load balancing module communicatively coupled to the plurality of physical hosts and] receive the virtual server resource modification signal and to determine whether the first physical host is overloaded and, in response to a determination that the first physical host is overloaded, to send a physical host transfer signal that indicates a second physical host *See also* construction of the term "the first physical host is overloaded" Structure: Physical Host Load Balancing Module 130 as described in '726 Patent, 5:21-65; 6:4-19; 10:53-11:52; Figure 5.	Not subject to § 112 ¶ 6 - in the alternative Function receive the virtual server resource modification signal and determine whether the first physical host is overloaded and in the case that it is send a physical host transfer signal indicating a second physical host Structure Dynamic Resource Configuration Module 100; Physical Hosts 160A-C; Virtual Servers 162A-G

The Court should construe this language which recites "a load balancing module … configured to … [perform a function]" as subject to 35 U.S.C. § 112 ¶ 6. The term "a load balancing module" is a term that fails to carry a structural meaning. Ex. 2 (Snoeren Decl.) ¶¶ 70-71; *Williamson*, 792 F.3d at 1350 ("It replaces the term 'means' with the term 'module' and recites three functions performed by the 'distributed learning control module.' 'Module' is a well-known nonce word that can operate as a substitute for "means" in the context of § 112, para. 6."). Furthermore, as discussed above, claim language reciting what the component is "configured to" do is functional. *MTD Prod. Inc.*, 933 F.3d at 1343.

Furthermore, for the reasons discussed in Section III.D, the term "program code for [performing a function]" is a means-plus-function term subject to 35 U.S.C. § 112 ¶ 6. Ex. 2 (Snoeren Decl.) ¶ 72.

Here, the '726 patent identifies a software algorithm that is associated with performing the claimed function. Figure 5, 5:21-65, 6:4-19, and 10:53-11:52 of the '726 patent, describe a component called a "Physical Host Load Balancing Module" (or "Physical Host Load Balancer") which "periodically monitors the resource usage of a group of physical hosts [] and transfers virtual servers to different ones of these physical hosts [] in order to balance the resource loads between the physical hosts []." '726 patent, 10:56-60.

The "Physical Host Load Balancing Module" is disclosed with a specific algorithm that involves receiving a signal from the virtual server resource modifier, using a load-balancing calculator to process this signal and a signal regarding the current physical host's resource loads, determining if the physical host is overloaded, and identifying a different available physical host using an easiest fit heuristic algorithm. As this is the only disclosure in the patent—identified by either side—of an algorithm for performing the claimed function, the Court should adopt VMware's proposal. Ex. 2 (Snoeren Decl.) ¶¶ 73-78.

G. "a dynamic virtual server mover [communicatively coupled to the plurality of physical hosts and] configured to receive the physical host transfer signal and, in response to the physical host transfer signal, to transfer the virtual server from the first physical host to the second physical host" ('726 patent claims 1 and 5) // "program code for creating a dynamic virtual server mover communicatively coupled to the plurality of physical hosts and configured to receive the physical host transfer signal and, in response to the physical host transfer signal, to transfer the virtual server from the first physical host to the second physical host" ('726 patent claim 4)

VMware Proposal	IV Proposal
Means-plus function terms. Function: [creating a dynamic virtual server mover communicatively coupled to the plurality of physical hosts	Not subject to § 112 ¶ 6 - in the alternative

and] receive the physical host transfer signal and, in response to the physical host transfer signal, to transfer the virtual server from the first physical host to the second physical host *See also* construction of the term "virtual server" Structure: Dynamic Virtual Server Mover 140 as described in '726 Patent, 11:63-12:23; Figure 6.	Function receive the physical host transfer signal and transfer the virtual server from the first physical host to the second physical host Structure Dynamic Resource Configuration Module 100; Physical Hosts 160A-C

The Court should construe this language which recites "a dynamic virtual server mover … configured to … [perform a function]" as subject to 35 U.S.C. § 112 ¶ 6. The term "dynamic virtual server mover" is a term that fails to carry a structural meaning. Ex. 2 (Snoeren Decl.) ¶¶ 79-80. Indeed, a search of all U.S. Patents shows that the term appears in only 3 patents (the '686, '726, and '937 patents). *See* Ex. 6. Furthermore, as discussed above, claim language reciting what the component is "configured to" do is functional. *MTD Prod. Inc.*, 933 F.3d at 1343.

Furthermore, for the reasons discussed in Section III.D, the term "program code for [performing a function]" is a means-plus-function term subject to 35 U.S.C. § 112 ¶ 6. Ex. 2 (Snoeren Decl.) ¶ 81.

The '726 patent identifies a software algorithm that is clearly associated with performing the claimed function. Figure 6 and the corresponding description at '726 patent, 11:63-12:23, describe a component called a "Dynamic Virtual Server Mover" "for transferring a virtual server from one physical host to another physical host." '726 patent, 11:63-65.

As shown in Figure 6 and described in '726 patent, 11:63-12:23, the "Dynamic Virtual Server Mover" is disclosed with a specific algorithm that involves storing state information, stopping virtual server processes in the first host, accessing the state information by the second host, and then starting virtual server processes on the second host. The '726 patent discloses the mover "may use either 'make, then break' timing or 'break, then make' timing for the transfer process." '726 patent, 12:21-23. Given that this is the only disclosure in the patent—identified by either side—of an algorithm for performing the claimed function, the Court should adopt VMware's proposal. Ex. 2 (Snoeren Decl.) ¶¶ 82-88.

H. "the dynamic virtual server mover is further configured to direct the first physical host to store, in the file system, a set of system files for the virtual server and to direct the second physical host to access, from the file system, the set of system files for the virtual server, thereby transferring the virtual server from the first physical host to the second physical host" ('726 patent claims 3 and 7)

VMware Proposal	IV Proposal
Means-plus function terms. Function: direct the first physical host to store, in the file system, a set of system files for the virtual server and to direct the second physical host to access, from the file system, the set of system files for the virtual server, thereby transferring the virtual server from the first physical host to the second physical host *See also* construction of the term "virtual server" Structure: Dynamic Virtual Server Mover 140 as described in '726 Patent, 11:63-12:23; Figure 6.	Not subject to § 112 ¶ 6 - in the alternative Function direct the first physical host to store a set of system files for the virtual server in the file system and to direct the second physical host to access the set of system files for the virtual server from the file system and transferring the virtual server Structure Dynamic Resource Configuration Module 100; Physical Hosts 160A-C; Virtual Servers 162A-G

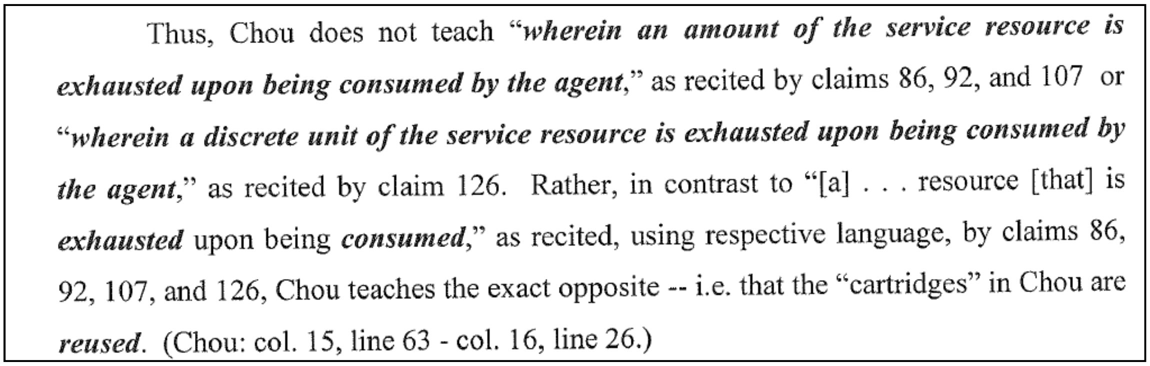
For the reasons discussed in Section III.F, the term "dynamic virtual server mover … configured to … [perform a function]" is a means-plus-function term subject to 35 U.S.C. § 112 ¶ 6. Furthermore, for the reasons discussed in Section III.F, the '726 patent identifies a software algorithm that is associated with performing the claimed function in Figure 6 and 11:63-12:23. Ex. 2 (Snoeren Decl.) ¶ 89.

IV. DISPUTED TERMS FROM U.S. PATENT NO. 7,949,752 (the "'752 patent")

A. "exhausted" ('752 patent claims 1, 9 and 24)

VMware Proposal	IV Proposal
"unavailable for reuse"	"used up to the allotted or pre-determined amount"

The disputed claim term is part of a claim limitation that recites: "wherein an amount of the ***service resource is exhausted*** upon being consumed by the network-based agent." '752 patent at claims 1, 9 and 24 (emphasis added).[6] VMware's proposal is supported by a clear disclaimer in the prosecution history where the applicant distinguished prior art having ***reusable*** resources as "the exact opposite" of the claim language:



> Thus, Chou does not teach "***wherein an amount of the service resource is exhausted upon being consumed by the agent***," as recited by claims 86, 92, and 107 or "***wherein a discrete unit of the service resource is exhausted upon being consumed by the agent***," as recited by claim 126. Rather, in contrast to "[a] . . . resource [that] is ***exhausted*** upon being ***consumed***," as recited, using respective language, by claims 86, 92, 107, and 126, Chou teaches the exact opposite -- i.e. that the "cartridges" in Chou are ***reused***. (Chou: col. 15, line 63 - col. 16, line 26.)

Ex. 14 (2010-08-20 Response) at 10 (emphasis in original). As shown, the applicant did not state that the prior art disclosed something ***unrelated*** to the claim term. Instead, the applicant's argument defines the claim term "exhausted" as "the exact opposite" of resources that can be "reused."

VMware anticipates that IV will argue that the prosecution history shows that the applicant abandoned this disclaimer. Not so. As shown in the summary of the prosecution history provided below, the applicant made this disclaimer and did not abandon the argument.

[6] For context, '752 patent provides an example that a "***service resource 25*** related to telephony services (e.g., voice mail and call placement) ***may comprise units of long-distance calling time which are consumed as an agent 22 places one or more calls*** utilizing such services 24." '752 patent at 11:8-12 (emphasis added).

1. Prosecution History Summary

The applicant first sought to broadly claim "using a service and a service resource." Ex. 9 (2009-03-16 Response) at claim 86. However, the examiner found that U.S. Patent No. 7,043,532 ("Humpleman") anticipated these broad claims. Ex. 10 (2009-05-05 Office Action) at 2–3.

The applicant then narrowed the claims to recite "using a service and a service resource configured to be consumed by the agent", and distinguished the prior art as merely using service resources. Ex. 15 (2009-08-05 Response) at 10 ("Humpleman describes that "[e]ach server device may include ***hardware as a resource*** in the network for providing services to the user" . . . not the claimed "means, including the network-based agent, for using a service and a service resource ***configured to be consumed*** by the agent when performing the operation on behalf of the user") (emphasis in original). However, in the next office action, the examiner maintained the prior art rejection. Ex. 11 (2009-10-30 Office Action), 2-3.

The applicant then narrowed the claims further to add "wherein the service resource is exhausted after it is consumed by the agent." Ex. 12 (2009-11-13 Response) at claim 86. The applicant distinguished the prior art by stating "in Humpleman 'hardware' is used as a resource, and is not 'consumed' or 'exhausted'". *Id.* at 10. The examiner responded by withdrawing the Humpleman objection and imposing a new prior art rejection involving U.S. Patent No. 6,247,056 ("Chou"). Ex. 13 (2010-03-16 Office Action) at 4–7. Chou disclosed using reusable software modules called "cartridges." *Id.* at 5.

The applicant then made minor claim amendments and distinguished the prior art as disclosing ***reusable*** resources, which the applicant called the "***exact opposite***" of the "***exhausted***" claim language. Ex. 14 (2010-08-20 Response) at 10 ("in contrast to '[a] ... resource [that] is exhausted upon being consumed,' as recited, . . . ***Chou teaches the exact opposite*** -- i.e. that the 'cartridges' in Chou are ***reused***") (emphasis in the original).

The examiner responded by calling the applicant's argument "misleading" because the '752 patent application also discloses reusable resources (e.g., memory storage space). Ex. 16 (2010-11-08 Office Action) at 2. However, although the examiner did not necessarily agree with the applicant's interpretation, the examiner withdrew its prior art rejection and imposed a new objection involving U.S. Patent No. 5,740,231 ("Cohn"). Cohn disclosed consuming ***non-reusable*** fees for using a service (e.g., similar to the '752 patent application's description, at 11:8-12, of consuming long distance minutes). Ex. 16 (2010-11-08 Office Action) at 3.

The applicant responded by "travers[ing]" and not "acquiescing" to the rejection. Ex. 17 (2011-01-04 Response) at 10. Instead, the applicant amended the claims and argued that the claims as amended "recite ***further features*** that distinguish over the applied reference." *Id.* (emphasis added). The claims were allowed in the following office action.

2. Argument

The prosecution disclaimer here could not be any clearer. The applicant stated that it is the "exact opposite" of the "exhausted" claim language when resources can be "reused". This argument was made specifically for purposes of distinguishing prior art. The applicant never abandoned this disavowal. The Court should adopt VMware's proposal. *See SanDisk Corp. v. Memorex Prods., Inc.*, 415 F.3d 1278, 1286 (Fed. Cir. 2005) ("When the patentee makes clear and unmistakable prosecution arguments limiting the meaning of a claim term in order to overcome a rejection, the courts limit the relevant claim term to exclude the disclaimed matter.").

In a prior litigation involving the '752 patent, the Magistrate Judge provided a report and recommendation construing the "exhausted" to be "used up to the allotted or pre-determined amount," opining that the "prosecution history, when considered as a whole, demonstrates that the patentee did not make a clear and unmistakable disclaimer." Ex. 18 (Magistrate Report) at 33-34. IV had cited to *Ecolab, Inc. v. FMC Corp.*, 569 F.3d 1335 (Fed. Cir. 2009) for the proposition that there is no disavowal when an applicant abandons a limiting argument after it was rejected by an examiner. *Id.* at 32. The Magistrate's finding was in error and *EcoLab* is easily distinguishable.

EcoLab involved a unique situation where the applicant made a ***legally incorrect*** argument. The applicant had argued that claims reciting opened-ended "consists essentially of" language encompassed "a single biocide" and not "mixtures of biocides." *Ecolab*, 569 F.3d at 1343. The examiner corrected the legal mistake, and the applicant responded by cancelling the claims without disagreement. *Id*. Under the *EcoLab* facts, it is unclear whether applicant was making a narrowing statement or if it merely misunderstood the law. The *EcoLab* court concluded that "a reasonable reader of this prosecution history could conclude that [applicant's] initial statements . . . were hyperbolic or erroneous, that the Examiner corrected [applicant's] error . . ., that [applicant] recognized its error and never again repeated or relied upon the erroneous rationale . . . ." *Id.*

The '752 patent's prosecution history, in contrast, presents no uncertainty or abandonment. There is no legal misstatement here—instead, the applicant made a purely ***factual*** statement that the "exhausted" limitation is "the exact opposite of" the prior art's "reusable" resources. Additionally, the '752 patent applicant did not cancel the claims without disagreement when the examiner called its arguments "misleading"—instead, the applicant responded by stating that it "respectfully traverse[s] this rejection," and "[w]ithout acquiescing to the propriety of the rejection . . . [made amendments] to recite ***further features*** that distinguish over the applied reference." Ex. 16 (2010-11-08 Response to Final Office Action) (emphasis added).

The present case is also distinguishable because, although the examiner stated that the applicant's arguments were misleading, the examiner accepted the disavowal by shifting its prior art rejection to a new reference (Cohn) that disclosed non-reusable resources. Furthermore, "[i]t is of no moment whether the examiner agreed with the disclaimer." *See, e.g., Uship Intellectual*

Properties, LLC v. United States, 714 F.3d 1311, 1315-16 (Fed. Cir. 2013) ("The analysis focuses on what the applicant said, not on whether the representation was necessary or persuasive: 'Regardless of the examiner's motives, arguments made during prosecution shed light on what the applicant meant by its various terms.'"). The key issue is whether or not the patentee rescinded the disclaimer.

Under the present facts, the applicant has not clearly rescinded the disclaimer of reusable resources. *See, e.g., Hakim v. Cannon Avent Group PLC*, 479 F.3d 1313, 1317-18 (Fed. Cir. 2007) ("Although a disclaimer made during prosecution can be rescinded, permitting recapture of the disclaimed scope, the prosecution history must be sufficiently clear to inform the examiner that the previous disclaimer, and the prior art that it was made to avoid, may need to be revisited."). It is of no moment that applicant amended the claims after the examiner called its arguments misleading. *See Desper Prods. Inc. v. QSound Labs. Inc.*, 157 F.3d 1325, 1335-36 (Fed. Cir 1998) ("That the prosecution shifted to a different focus does not blunt the impact of those remarks made to overcome the prior rejection."). IV's proposal attempts to recapture claim scope which was disavowed by the original applicant and therefore should be rejected.

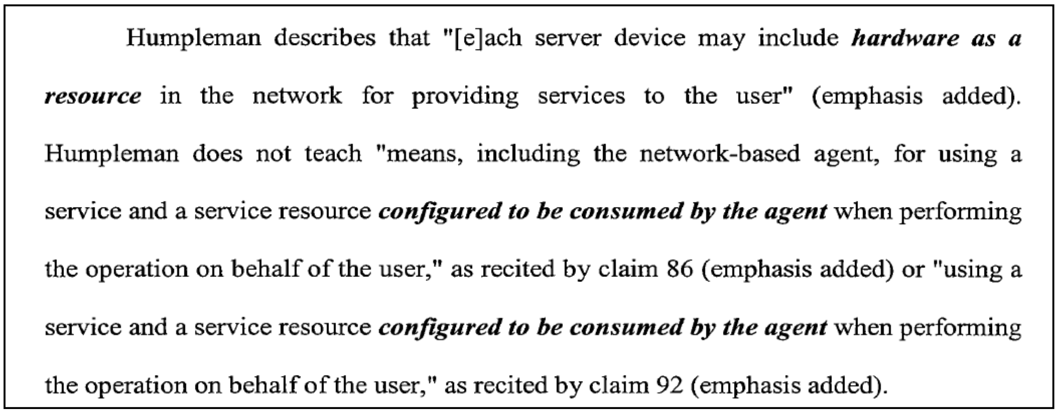
B. "consumed" (recited in '752 patent claims 1, 9 and 24)

VMware Proposal	IV Proposal
"used up"	"used"

VMware's proposal for this term is supported by narrowing statements made during prosecution and a definitional statement provided in the patent specification.

During prosecution, the '752 patent applicant first tried to claim mere ***use*** of resources. *See, e.g.,* Ex. 9 (2009-03-16 Response) at claim 86 ("means, including the network-based agent, ***for using*** a service and ***a service resource***"). But later, in response to a prior art rejection, the applicant was forced to narrow the claims to recite that the resource is "configured

to be consumed by the agent." Ex. 15 (2009-08-05 Response) at claim 86. In making this amendment, the applicant distinguished prior art with an identified agent (a GUI) configured to merely ***use*** resources (hardware) for providing a service:



Humpleman describes that "[e]ach server device may include ***hardware as a resource*** in the network for providing services to the user" (emphasis added). Humpleman does not teach "means, including the network-based agent, for using a service and a service resource ***configured to be consumed by the agent*** when performing the operation on behalf of the user," as recited by claim 86 (emphasis added) or "using a service and a service resource ***configured to be consumed by the agent*** when performing the operation on behalf of the user," as recited by claim 92 (emphasis added).

Ex. 15 (2009-08-05 Response) at 10 (emphasis in original). As evidenced by this history, "consumed" cannot simply mean "use" as IV proposes—the claim language was amended to employ both "using" and "consumed" in the same phrase, intending that the two words have different meaning. The applicant then used this narrowing amendment to argue around prior art that disclosed an agent configured to use a resource.

Indeed, the past tense use of the term "consumed" is substantively different than "consume" as used throughout the patent. "Consumed" indicates that a resource was used completely or "used up" to some allotted amount, whereas "consume" refers to the present tense and ongoing "use" of a resource. For instance, "[e]ntities" may "purchase, ***consume, or otherwise use*** services." *See* '752 Patent at 2:2-3 (emphasis added). On the other hand, a "service wrapper 26 ***may maintain a record of the amount of service resource 25 consumed*** by various agents." *Id.* at 12:9-12 (emphasis added). Here, "consume" clearly refers to consuming resources that are still available, while "consumed" refers to a resource that was used up to a certain amount. *See also id.* at 22:30-33 ("each service permission 64 specifies whether the agent 22 is authorized ***to***

consume a particular service resource 25 and, in some instances, the amount of such service resource 25 ***that is allowably consumed*** by that agent 22.").

Indeed, the '752 patent's specification defines how "consumed" is different than mere "use". Specifically, the '752 patent equates the term "consumed" with the term "used up". *See* '752 Patent at 8:21-23 (resources "may be ***'consumed' or 'used up'*** during the operation of network system 2") (emphasis added). This is a definitional statement defining "consumed" as VMware proposes. *See, e.g., Intellectual Ventures I, LLC v. Lenovo Group Ltd.*, 365 F.Supp.3d 200, 206 (D.Mass. 2019) ("A term set off by quotation marks is 'often a strong indication that what follows is a definition.'") (quoting *Sinorgchem Co., Shandong v. Int'l Trade Comm'n*, 511 F.3d 1132, 1136 (Fed. Cir. 2007)). This definitional statement is consistent with common usage of the term "consumed." *See* Ex. 27 (Webster definition of consume) (defining as "to do away with completely," "use up," and "to engage fully"). This definition is further supported by the specification's only other use of the term consumed in quotations—where the specification describes that "[a]t least some of service resources 25 may comprise discrete units which are 'consumed' during utilization of the respective resource by an agent 22 . . . [such as] units of long distance calling time". '752 patent, 11:5-12.

IV's proposal is shown improper when the full claim limitation is rewritten replacing the term "consumed" with "used" and "exhausted" with "used up to the allotted or pre-determined amount" as IV proposes. Claim 1 would become incomprehensible, reciting: "wherein an amount of the service resource is ~~exhausted~~ used up to the allotted or pre-determined amount upon being ~~consumed~~ used by the network-based agent" (i.e., an amount of service resource would not be "used up" simply *upon* being used). This issue becomes even clearer when read in the context of "service resources" as defined in the specification. One example of "services resources" is "disk space for storing e-mail messages." '752 patent at 11:2-5. Disk space is not "used up to the allotted or pre-determined amount" simply upon being "used."

In the prior litigation, the Magistrate Judge opined that "consumed" means "used" and not "used up." *See*, Ex. 18 (Magistrate Report) at 29–30 (arguing that the '752 patent uses "consumed" synonymously with "used"). The Magistrate's finding was in error. Many of the examples cited by the Magistrate simply refer to embodiments where the term "consumed" is included in a list of alternatives. *See, e.g., id*. at 30 ("computational resources 21 which may be expended, consumed, or used during the operation"). None of the examples overcome the narrowing statement made during prosecution or the specification's definitional statement identified above.

C. "service" ('752 patent claims 1, 3, 9 and 24)

VMware Proposal	IV Proposal
"An application that is used by an agent on behalf of a principal"	"Network functionality available to agent(s)/network-based agent(s)"

The Court should adopt VMware's proposal for this term. The specification makes clear that an "agent uses the service on behalf of a principal." *See, e.g.,* '752 Patent at 3:3-5. In the section titled "Services" the specification further states that the services may "comprise one or more software applications providing various capabilities that are available to a principal." *Id.* at 10:17-19. The next sentence explains that "[e]ach service 24 may be utilized by one or more agents 22 in order to perform their respective tasks." *Id.* at 10:19-21.

IV's proposal, in contrast, does not help clarify the claim scope. The relevant claims already recite execution by a "network-based agent", therefore IV's language of "network functionality" adds nothing. Therefore, VMware's proposed construction should be adopted.

D. Means-Plus-Function Terms

Attached as Ex. 28 is a list of claim terms for which the parties dispute only whether citations to the specification should be included in what is provided to the jury. VMware contends that compliance with 35 U.S.C. § 112 dictates a construction of means-plus-function claims that includes a cite to the specific portion of the specification where the corresponding structure is disclosed. Section 112 states, in relevant part: an element in a claim for a combination may be expressed as a means or step for performing a specified function without the recital of structure, material, or acts in support thereof, and such claim shall be construed to cover ***the corresponding structure***, material, or acts ***described in the specification*** and equivalents thereof. 35 U.S.C. § 112 (emphasis added). As such, the Court should adopt VMware's proposals for the means-plus-function elements. *See Ariba Inc. v. Emptoris, Inc.*, No. 9:07-cv-90 (E.D. Tex. Aug. 7, 2008) (construing means-plus-function terms with citations to the specification); *see also Personal Audio,* 2011 WL 11757163, at *11 (construing means-plus function terms the same way).

Additionally, a dispute remains as to the structure for the means-plus-function term "means for monitoring an amount of the service resource used by the network-based agent" of claim 4:

VMware Proposal	IV Proposal
Function: monitoring an amount of the service resource used by the network-based agent Structure: monitor as described in '752 patent at 16:50-61	Function: monitoring an amount of the service resource used by the network-based agent Structure: Service Wrapper 26

The parties agree on the function for the means term. The disagreement lies in the proposed structure. VMware's proposal is correct. The specification states that the "[m]onitor 50 generally ***functions to monitor the amount of respective service resources*** 25 expended, ***used***, or otherwise consumed ***by one or more agents*** 22 which have been authorized to access the service 24." *See,* '752 patent at 16:50-61 (emphasis added). This disclosure matches the function proposed by both parties. In contrast, the structure that IV points to, the Service Wrapper 26, is disclosed as comprising the monitor ***in addition to other structure and functionality.*** *See, id.* at 16:28-29 (comprises both a "converter 48" and a "monitor 50." *See, id.* at 16:28-61. The converter is irrelevant to the function of monitoring service resources, as it is generally used to "convert between a computer language (or instruction set) used within agent server 20 and a computer language (or instruction set) used within the respective service 24." *See, id.* at 16:30-33. Therefore, VMware's proposed construction should be adopted.

V. DISPUTED TERMS FROM U.S. PATENT NO. RE43,051 (the "'051 patent")

A. "virtual server" ('051 patent claims 1, 3 and 6)

VMware Proposal	IV Proposal
"a process executing on a host computer that accept communications requests"	Plain and ordinary meaning, or alternatively: "virtual machine(s) that reside(s) on a physical server and use(s) the physical server's resources but has/have the appearance of being a separate dedicated machine(s)"

The parties dispute whether IV can change the meaning of virtual server to address an entirely different technology (virtual machines) that is not discussed anywhere in the intrinsic record, when the intrinsic record instead consistently describes the virtual servers of the invention as relating to processes, like server applications. VMware's proposal is necessitated by the intrinsic record and finds additional support in the extrinsic evidence. IV's proposal disregards critical intrinsic evidence, and instead relies completely on extrinsic evidence to redefine virtual server to mean a virtual machine, which IV's evidence says is software that mimics the performance of a hardware device. Worse, IV selectively modifies the dictionary definition it relies upon to arrive at this proposed construction. A fundamental dispute between the parties is whether a POSITA would have understood "server" to be a process, or a machine. For the reasons explained below, the intrinsic record is unequivocal that a POSITA would have understood that a "server" in the context of the intrinsic evidence refers to a process, not a machine, and that the term virtual server means "a process executing on a host computer that accept communications requests."

The claim language supports VMware's proposal. Claim 1 describes "a host computer containing a plurality of virtual servers," and claim 3 describes "a host computer containing multiple virtual servers." A POSITA would have understood the mechanism for placing multiple virtual servers on a single host machine, as described in claims 1 and 3, involves executing processes on the host machine. *See* Ex. 2, ¶ 91; Ex. 19 at 39 (describing the Apache web server, which is comprised of "processes to handle client requests"); Ex. 20 at 45 ("Virtual servers are servers that have different addresses but refer to the same Web server.").

For instance, a POSITA would have been familiar with the use of virtual servers to host multiple websites on a single host machine: "The illusion of more web sites existing than actual web servers is created through the use of ***virtual servers*** (hosts). Web hosting service is based on this technique. Web hosting service uses the possibility to create a set of ***virtual servers*** on the same server." Ex. 21 at 14–15 (emphasis added); Ex. 2, ¶¶ 92–97. An important example is Apache, "the most widely used web server for commercial web sites." Ex. 19 at xviii; *see also* '051 patent at 3 (citing to Ex. 29, BEN LAURIE AND PETER LAURIE, APACHE: THE DEFINITIVE GUIDE (1999)). "Apache is an example of a *preforking* server," which "means that the main server starts a pool of processes to handle client requests, rather than forking a new process for each incoming request." Ex. 19 at 39; Ex. 2, ¶¶ 94–97. Each of a plurality of Apache processes running on a host computer can be bound to a specific IP address, in which case each process is a virtual server. In this configuration, "[a]ll connections to the Apache server on this IP address are handled by the ***virtual server*** for this site, which might be one of many virtual sites being hosted on the same [physical] server." *Id.* at 98 (emphasis added); *see also* Ex. 29 at 163, 177, 242–43, 295 (describing how to configure Apache virtual severs); Ex. 2, ¶¶ 93–95.

The specification reinforces, and in fact requires, an understanding that the term virtual server in claims 1 and 3 refers to processes, like those used in the Apache sever application. Ex. 2, ¶¶ 96–97. The '051 patent explains that "a server application executing on a single physical host can be programmed to process requests made to multiple network addresses." '051 patent at 2:47–21. The specification discloses that the '051 patent "is related to U.S. patent Ser. No. 09/452,286, entitled 'Providing Quality of Services Guarantees to Virtual Hosts,'" which "is incorporated by reference herein in its entirety." *Id.* at 1:16–21. The patentee further provides that the '286 application describes how to create the private virtual servers disclosed in the '051 patent. *Id.* at 3:64–67. The '286 application ultimately issued as U.S. Patent No. 6,976,258.

The '258 patent only describes how to create and configure processes, like the Apache virtual servers. *See* Ex. 2, ¶¶ 96–97. The '258 patent defines "[a] server" as "a process, executing on a dedicated physical services client [that] services client requests for a single network address (physical host) only." '258 patent at 3:23-25. "A server program executing on the host opens a communication transport channel (socket) and allows receipt of incoming communications targeted for any of the multiple network addresses assigned to the host. . . . However, multiple communication requests made to a plurality of network addresses associated with a single physical host require simultaneous service. For this reason, a virtual host server typically accepts the communications requests itself and creates child processes to service the requests." *Id.* at 1:24–44. In other words, the virtual server described in the '258 patent (and thus incorporated into the '051 patent) refers to a server application that uses child processes to service communication requests on a plurality of network addresses. The '258 patent explains that "[t]wo known methods exist for utilizing child processes to service communication requests." *Id.* at 1:45–2:5. The two methods described are "fork after accept" (generating child processes after a request is received), *id.* at 1:45-55, and, as with Apache's virtual servers, *see* Ex. 19 at 39, "fork before accept" (generating child processes in advance to service requests), '258 patent at 1:56–5; Ex. 2, ¶ 96–97.

Whichever method is used to create them, the ’258 patent teaches that these processes can be bound to individual network addresses, *see id.* at 2:6–22, which “allows an ISP to utilize one physical host computer to provide host services to multiple customers.” *Id.* at 2:38–40. Moreover,

> [b]y utilizing the fork before accept method or the fork after accept method, the virtual host server can service requests to multiple network addresses or domain names. Thus, the functionality of numerous hosts is provided by a single physical host computer, servicing requests made to a plurality of network addresses and domain names by multiple customers.

Id. at 2:40–52, 1:24-44, 1:45-2:5, 3:23-24, 4:36-49. This background—which the patentee incorporated by reference to explain how to create the virtual servers in the ’051 patent—provides clear context from which a POSITA would understand that the virtual server of the ’051 patent is “a process executing on a host computer that accepts communications requests.”

All of these disclosures contradict IV’s contention that virtual server means a virtual machine. *See* Ex. 2, ¶ 98. Despite the robust discussions of processes and how to use them as virtual servers in the intrinsic record, IV has produced only a single dictionary definition (from after the priority date of the patent) to justify redefining virtual server as virtual machine. That same reference defines a virtual machine as “[s]oftware that mimics the performance of a hardware device such as a program that allows applications written for an Intel processor to be run on a Motorola chip.” *See id.* at 554. There is no support in the claims or the specification for IV’s contention that the virtual servers of the ’051 patent are software that mimics the performance of a hardware device. IV’s proposed definition contradicts the disclosures in the specification that the virtual servers relate to server applications that use child processes to service communications requests—a distinctly software-based solution, not hardware emulation. *See* Ex. 2, (Snoeren Decl.) ¶¶ 98–99. Given the intrinsic evidence, a POSITA would not have understood the ’051 patent to use virtual server to mean a virtual machine like that disclosed in IV’s extrinsic evidence. *Id.*

The prosecution history further confirms VMware's proposal. Ex. 2, ¶ 100. The '051 patent was a reissue of U.S. Patent No. 6,948,003. During prosecution of the '003 patent, the applicant responded to multiple rejections based on U.S. Patent No. 6,286,047 ("Ramanathan") (Ex. 25) and U.S. Patent No. 6,247,057 ("Barrera") (Ex. 26). *See, e.g.*, Ex. 22 at 14–15 (2003-10-14 Response). The applicant conceded that Ramanathan discloses virtual servers "in the context of determining whether a host supports virtual servers," and further explained that an example of a virtual server in Ramanathan is a "website hosted by a virtual server [that has] a unique IP address." *Id.* Likewise, the applicant tried to differentiate the virtual servers in Barrera because it "requires that the virtual services each have a unique locator ID." *Id.* The applicant never disputed that the virtual servers in Ramanathan and Barrera were the same as those in the application.

Ramanathan and Barrera both describe virtual servers like the Apache virtual servers described above. Ex. 2, ¶ 100. Ramanathan explains that "the virtual servers model" involves hosting "all of the customer websites . . . using a single host machine," Ex. 25 at 31:48–51, and provides example templates addressing Apache virtual servers. *Id.* at 11:44–13:30. Barrera similarly explains that "current server technology allows administrators to run multiple instances of the same service on a single machine." Ex. 26 at 2:1–12. As an example, Barrera also notes that it is common "for a Web server to support thousands of domains on the same Web service," such that "the physical host server can be said to support multiple 'virtual services' on multiple 'virtual servers.'" *Id.* at 2:13–19. Nothing in either Ramanathan or Barrera indicate that a virtual server is anything other than a process, much less software emulating a hardware device. Ex. 2, ¶ 100.

The intrinsic evidence speaks clearly, and the extrinsic evidence supports what it says—virtual server means a process executing on a host computer that accepts communications requests.

B. "physical interface[s]" ('051 patent claims 1 and 3)

VMware Proposal	IV Proposal

"hardware that provides a point of communication between two or more devices"	Plain and ordinary meaning

VMware contends the term "physical interface" needs to be construed to resolve an apparent dispute between the parties. It appears IV and VMware differ over whether the plain meaning of the word physical interface as it is used in the '051 patent means hardware. IV has refused to confirm this is actually in dispute, and has instead opaquely insisted that the term has a plain and ordinary meaning without providing any clarity as to what it believes that meaning is, or whether that meaning comports with VMware's understanding that the physical interface is hardware. As explained below, the proper construction of physical interface is "hardware that provides a point of communication between two or more devices."

The claim language requires VMware's construction. The term physical interface appears in independent claims 1 and 3 to facilitate communication between two or more devices. *See, e.g.*, '051 patent at claim 1 ("receiving . . . a transmission on a physical interface"; "sending the transmission . . . on the determined physical interface"). Notably, the patentee chose to use the term *physical* interface, not just interface. This choice has meaning. Ex. 2, ¶ 104. The term physical is used throughout the specification to differentiate hardware components from virtualized or logical components. For instance, the specification uses the term physical to differentiate between the physical host computers (or physical servers) and the virtual servers running on them. *See, e.g.*, '051 patent. at 2:47–64 ("a server application executing on a single physical host"), 4:35–38 ("one or more private virtual servers located on a physical host computer"), 4:56–65, 7:1–11, 7:47–48, 8:4–5. Likewise, the specification uses the term physical to separate the logical tunnel constructs from the hardware over which the data is transmitted. *See id.* at 10:17–20 ("Each tunnel is not a separate physical connection; it is a specific encapsulation of data allowing the data to be separated out from other data sent on a physical connection."). A POSITA would have thus understood the patentee's use of physical mean a hardware interface comprised of circuitry. Ex. 2, ¶¶ 102–06; *see also* Ex. 23, (IEEE definition: "circuitry that interfaces a module's nodes to the input link, output link, and miscellaneous signals").

The specification also provides examples of physical interfaces that confirm VMware's proposal, such as a hardware network interface cards. '051 patent at 8:51–57, 9:4–9, 9:56–60. So too do uncontested examples from the examiner in the prosecution history, like "physical copper connections." Ex. 24 (2010-05-17 Response) at 7. There is no indication in the intrinsic record that the physical interface described in the claims is anything other than hardware that provides a point of communication between two or more devices.

C. physical interfaces and tunnel identifiers in the storing / receiving / determining / sending terms ('051 patent claims 1 and 3)

The terms "physical interface(s)" and "tunnel identifier(s)" are each used six times in each independent claim of the '051 patent. Because the claim language is ambiguous about which of the physical interfaces and tunnel identifiers are being referenced at the various steps, this creates a significant antecedent basis problem that makes the claims impossible to decipher without clarification. VMware's proposal for these terms (*see* Ex. 8) offers a simple way to resolve this problem that is fully supported by the specification—identifying each iteration of "physical interface(s)" and "tunnel identifier(s)" as either corresponding to an incoming (received) transmission or an outgoing (sent) transmission.

Claims 1 and 3 use two distinct sets of physical interfaces and tunnel identifiers. One set is of physical interfaces and tunnel identifiers associated with the receiving step; these are the physical interfaces and tunnel identifiers associated with the incoming transmission. When a transmission is received, the method requires identifying from the customer lookup table (or customer lookup information) the customer forwarding table (or customer forwarding information) that is associated with the pair of the physical interface and the tunnel identifier of the incoming transmission. The method then requires looking in the identified customer forwarding table (or customer forwarding information) for the destination network address of the incoming transmission to determine the pair of a physical interface and a tunnel identifier to use for sending the transmission. The transmission is then sent using the physical interface and tunnel identifier determined from the customer forwarding table (or customer forwarding information).

The physical interfaces and tunnel identifiers stored in the customer lookup table (or customer lookup information) and used in the receiving step are the *incoming* physical interfaces and the *incoming* tunnel identifiers. Figure 8 shows the customer lookup table of the invention, which includes the "Incoming Physical Interface" and the "Incoming Tunnel ID," and which are together associated with a "Customer ID." The specification further explains that the fields in the customer lookup table include the "incoming physical interface," the "incoming tunnel identifier," the "customer identifier," and that "[t]his customer identifier provides an index to the correct customer forwarding table associated with the physical interface/tunnel identifier pair." '051 patent at 11:64–12:1, 12:59–63 ("The incoming physical interface and tunnel identification information is read 740 from the packet 718, and presented 744 to a customer lookup table.").

Similarly, the physical interfaces and tunnel identifiers stored in the customer forwarding table (or customer forwarding information) and used in the sending step are the *outgoing* physical interfaces and the *outgoing* tunnel identifiers. Figure 9 shows the customer forwarding table of the invention, which includes the "Outgoing Tunnel ID" and the "Outgoing Physical Interface," which are together associated with a "Destination IP Address." The specification teaches that the fields in the customer forwarding table includes the "destination IP address," the "outgoing tunnel identifier," and the "outgoing physical interface," and that "[b]ased upon the destination IP address of the particular transmission, the proper outgoing tunnel and outgoing physical interface is determined." *Id.* at 12:10–20, 11:30–32, 11:44–31, 12:30–33, 13:5–11 ("From the information contained in customer forwarding table 910, the correct outgoing physical interface and tunnel identifier for packet 718 is identified 754."), 14:56–63. VMware's proposals are correct because they use clear disclosures in the specification that would inform the understanding of a POSITA to correct the antecedent basis issues that would otherwise render the claim language indefinite.

VI. DISPUTED TERMS FROM U.S. PATENT NO. RE44,818 (the "'818 patent")

A. "hierarchical token bucket resource allocation" (recited in '818 patent claims 1, 17, 30, 32 and 42)

VMware Proposal	IV Proposal
the specific class-based scheduling algorithm known in the art as the "hierarchical token bucket"	Plain and ordinary meaning

VMware's construction is the only meaning that any POSITA would ascribe the term. The terms "hierarchical token bucket" and "token" as used in the claims of the '818 patent refer to the very specific method of allocating bandwidth resources referred to in the art as the "hierarchical token bucket," or "HTB." Ex. 2, Snoeren Decl. ¶¶ 107-112. HTB was invented or at least popularized by Martin Devera when he implemented it as a queuing discipline in the Linux operating system. *Id.* at ¶ 107. It is not something that the inventors created. In fact, during the prosecution, the inventors submitted an article[7] describing the well-known HTB algorithm in information disclosure statements. *See* '789 Patent File History, 2008.01.15 Information Disclosure Statement; '818 Patent File History, 2013.08.04 Information Disclosure Statement. This is one of many prior art articles (five of which appear on the face of the patent) that demonstrate that HTB is a "proper noun" or "term of art" that refers specifically to the scheduling mechanism known as "hierarchical token bucket" and not anything else. Ex. 2, ¶ 107.

[7] Ex. 43, ("Valenzuela Article").

Token buckets are based around the concept of exchanging logical "tokens" for the right to send data. Tokens are stored in buckets, which are replenished (and drained) at parameterized rates. Dr. Snoeren explains how the HTB uses a specific hierarchical arrangement of token-bucket rate limiters, each of which are a basic mechanism used for traffic management in computer networking. Ex. 2, ¶ 110. In HTB, traffic is divided into classes, and each class is associated with a token bucket in the hierarchy. *Id*. The tokens may be "borrowed" from buckets higher up in the hierarchy in accordance with a well-understood process. *Id*.

IV's position seems to be that the term does not have a specific meaning, but that it could refer to any hierarchical structure for managing bandwidth. Dr. Snoeren explains HTB was one of several well-known and frequently implemented class-based techniques to manage bandwidth resources, each of which generally had its own proper name and each of which was understood to be different. *Id*. at 108.

Dr. Snoeren confirms that the inventors used the term HTB in this customary way and did not intend or attempt to deviate from commonly understood meanings. *Id*. at ¶ 111. The patent itself uses "HTB" as a proper noun and notes that HTB it is one specific form of scheduling and that there are other, unclaimed methods. For instance, it states that "Hierarchical token bucket can be considered as a class based scheduling mechanism" and states that the "QoS manager" will "queue or forward" traffic "using scheduling and queuing methods **such as** hierarchal token bucket (HTB)." '818 patent at 9:51-10:53 (emphasis added). This description mirrors the description in the Velanzuella reference (cited by the '818 patent) and is consistent with how the term was understood in the art. Ex. 2, at ¶ 111. Moreover, during prosecution, applicants argued that the claims were patentable because the HTB was different than other "hierarchical or tree structure for storing resource reservations." Ex. 31 (2009.08.25 Response) at 12. IV disagrees with VMware's proposed construction but refused to say what was wrong with VMware's construction. Ex. 44. This does not resolve the dispute. VMware's construction should be entered. *See O2 Micro*, 521 F.3d at 1361; *US Foam Inc. v. On Site Gas Systems, Inc.*, 735 F. Supp. 2d 535, 556 (E.D. Tex. 2010) ("Even though the parties have not identified the substance of their position is or the real dispute, the Court must nonetheless fulfill its duty to determine the proper scope of the claims.").

B. "token" ('818 patent claims 1, 17, 30, 32-33, 37-42)

VMware Proposal	IV Proposal
"token as used in hierarchical token bucket resource allocation"	Plain and ordinary meaning

This Court should adopt VMware's proposed construction for the term "token" because it is consistent with how tokens were used in token-bucket traffic management approaches such as HTB at the time of the invention. Ex. 2, Snoeren Decl. ¶ 110. The intrinsic and extrinsic evidence discussed in section VI.A above applies here. In contrast, IV's proposed construction[8] is contrary to well-established Federal Circuit Precedent. *See O2 Micro*, 521 F.3d at 1360-6.

C. "enforc[e/ing]", "receiv[e/ing]", "classify[ing]", "compar[e/ing]", "forward[ing]", and "buffer[ing]" ('818 patent claims 1, 17, 30, 32, 33, 37, 38, 39, 42)

VMware Proposal	IV Proposal
"enforcing . . . across the physical [storage network] interface of the virtual I/O server" "receiv[e/ing] in the virtual I/O server" "classify[ing] in the virtual I/O server" "compar[e/ing] in the virtual I/O server" "forward[ing] in the virtual I/O server" "buffer[ing] in the virtual I/O server"	Plain and ordinary meaning (for each)

[8] In IV's disclosure of extrinsic evidence," IV cited to a dictionary definition for "token." However, a POSITA would understand that this definition is unrelated to the tokens used in the context of token-bucket traffic management approaches such as HTB. Ex. 2 (Snoeren Decl.) ¶ 110.

VMware's proposals for these terms "stays true to the claim language and most naturally aligns with the patent's description of the invention," and thus is the "correct construction." *Phillips*, 415 F.3d at 1316. The Federal Circuit has approved constructions that incorporate features that are "repeatedly and consistently" described as aspects of the invention. *See, e.g., SkinMedica, Inv. v. Histogen, Inc.*, 727 F.3d 1187, 1203-04 (Fed. Cir. 2013) ("clear, repeated, and consistent statements in the specification" limit claim scope); *Microsoft Corp. v. Multi-Tech Systems, Inc.*, 357 F.3d 1340, 1347-48 (Fed. Circ. 2004) (statements were "not limited to describing a preferred embodiment, but more broadly describe[d] the overall inventions").

Here, the patent discloses that "Fig. 4 is a component diagram showing the two-tier hierarchical components **<u>of the virtual I/O server</u>** QoS." '818 patent, 8:40-44 (emphasis added). And figures 5-10, repeatedly and consistently describe how the "receiv[e/ing]," "classify[ing]," "compar[e/ing]," "forward[ing]," and "buffer[ing]" occur in the virtual I/O server. For example, Figure 5 is a flow chart illustrating that these functions occur in the Virtual I/O server. *See* '818 patent Fig. 5 at 502, 504, 510, 516, 526, 528. Likewise, figures 6-10 also illustrate that these functions occur in the Virtual I/O server. In fact, nowhere else in the '818 patent does it describe these functions occurring anywhere else besides the Virtual I/O server.

The "enforc[e/ing]" term is also repeatedly and consistently described in the specification as being enforced across the physical interface of the Virtual I/O server.[9] Although the term "compar[e/ing]" does not explicitly appear in Figures 5-10, when reading the limitation in light of the claims, figures 5-10 refer to a comparison made between the received storage command,

[9] *See, e.g.,* '818 patent, 2:5-9 ("The hierarchy is based on partitioning of network interfaces and I/O subsystems transaction types, with QoS allocation decisions made on each hierarchy independently. This distributed transaction scheme provides scalable and fine-grain QoS management in virtual I/O servers."); *id.*, 8:5-28 ("QoS is performed on I/O communications from application servers 102 in various hierarchical tiers in virtual I/O server 106.").

input/output communication, or local area network packets and tokens which is shown throughout figures 5-10. *See, e.g.*, Fig. 5 at 510 (sufficient tokens to forward?). Moreover, the '818 patent makes clear that "Virtual I/O Server 106 provides the storage and external networking needs of application servers"'818 patent at 3:15-16. Accordingly, a POSITA would understand that the only relevant place for these functions to exist is in the virtual I/O server itself.

As a preliminary matter, the parties do not dispute the meaning "enforce[e/ing]," "receiv[e/ing]," "classify[ing]," "compar[e/ing]," "forward[ing]," and "buffer[ing]." However, as with other terms, IV has simply disagreed with VMware's proposals without providing a substantive explanation. Here, VMware's proposed constructions describe the features of the invention and limits the scope of the invention accordingly. *See Verizon.*, 503 F.3d at 1308. In contrast, IV's "plain and ordinary meaning" proposal failed to provide a meaning that would "assure that the jury understands that it is not free to consider its own meaning for disputed claim terms," thus, it's proposed construction should be rejected. *Sulzer*, 358 F.3d at 1366.

D. "maintaining a connection over a network fabric" ('818 patent claims 1, 17, 30, 32 and 42)

VMware Proposal	IV Proposal
"maintaining a connection between the physical interface of the application server and the physical interface of the virtual I/O server over a network fabric"	Plain and ordinary meaning

VMware's proposal is consistent with how a POSITA understands the term "network fabric." That is, a POSITA would understand that a physical interface connects to a network fabric. Ex. 2, Snoeren Decl. ¶¶ 113-14 ("In one implementation, the I/O fabric stack and I/O fabric PHY interface can employ the Reliable Connections (RC) supported by the Infiniband standard.") (citing '818 patent at 3:9-11). Indeed, the specification discloses that the application server connects to the network fabric and the virtual I/O server also connects to the network fabric. *See*

’818 patent at Fig. 1; *see also id.* at 3:22-23 (“One or more application servers 102 might be connected to the virtual I/O server 106 over I/O switch fabric 104”). As explained below, the intrinsic evidence supports the understanding of a POSITA that the term “connection” refers to the network fabrics connection with the physical interface of the application and the physical interface of the I/O switch fabric.

The specification specifically discloses that the physical interface of the application server connects to the network fabric. To illustrate, Figure 2 depicts the protocol stack of an application server. As shown in Figure 2 of the ’818 patent, the application server’s “I/O Fabric PHY interface 202” connects with the “I/O switch fabric 104”; *see also, e.g., id.* at 4:32-33 (“I/O fabric PHY interface 202 generally refers to the hardware interface or interconnection to the I/O switch fabric.”). Likewise, the physical interface of the virtual I/O device connects to the network fabric. To illustrate, the specification discloses that “Virtual I/O server 106 connects to the I/O switch fabric 104 through I/O fabric interface 110 such as Infiniband ports.” ’818 Patent at 3:9-11. A POSITA would have known that Infiniband ports are physical interfaces Ex. 2, Snoeren Decl. ¶¶ 113-14 (“Infiniband is a popular networking standard that would have been well known to one of skill the art, and understood to employ physical network interfaces.”).

IV disagrees with VMware’s proposed construction but will not provide VMware notice as to why. Instead, it proffers that the term should be afforded its “plain and ordinary meaning.” IV’s proposed construction should be rejected because “when the parties present a fundamental dispute as to the scope of the asserted claims, the Court, and not the jury, must resolve that dispute.” *O2 Micro*, 521, F.3d at 1360; see *also Reedhycalog UK, Ltd. v. Baker Hughes Oilfield Operations Inc.*, 2008 WL 2152268, at *1 (E.D. Tex. May 21, 2008) (“a court may not decline to construe a claim term or rely on the term’s ordinary meaning where such a construction does not resolve the parties' claim-scope dispute, allowing the parties to present claim scope arguments to the jury"). VMware's proposed construction should be accepted because the "statements describing the [the network fabric connections with the application server and Virtual I/O server as physical] found throughout the specification were not limited to describing a preferred embodiment, but more broadly describe[d] the overall invention." *Microsoft*, 357 F.3d at 1347.

For the reasons stated above, the Court should accept VMware's proposed construction.

E. "virtual storage network interface layer of an application server" / "virtual network interface layer of an application server"/ "virtual interface layer of an application server" ('818 patent claims 1, 17, 30, 32 and 42)

VMware Proposal	IV Proposal
"a virtual storage network interface to higher layers of the virtual node in an application server" / "virtual network interface layer to higher layers of the virtual node in an application server" / "virtual interface layer to higher layers of the virtual node in an application server"	Plain and ordinary meaning

The '818 patent describes the operation of the interfaces for the virtual nodes: "virtual network interface 220 presents a virtual link layer interface to higher layers of the protocol stack." '818 Patent at 5:16-17. VMware's proposed construction is consistent with the intrinsic record. For example, claims 1, 17, and 30 recite "the virtual storage network interface layer is associated with a virtual storage node identifier," and claims 32 and 42 recite "presenting, at a physical interface, a virtual node identifier to a local area network." From the specification, a POSITA would know that a virtual node resides on the application server. *See* '818 patent at 6:27-29 ("In another implementation, an application server is a virtual machine server, hosting one or more virtual machine monitors."). These virtual nodes located on the application servers must have a virtual network interface in order to communicate with the subsystems via the virtual I/O server. *Id.* at 3:25-28 ("Application servers 102 might include one or more virtual network interface modules to enhance the performance of their virtual access with SAN I/O subsystems 114 and

LAN I/O subsystems 116."); *see also id.* at 3:18-20 ("the virtual I/O server 106 creates virtual device interfaces for application servers 102."); Fig. 2 at 208a and 220. Networking protocols are typically described in terms of a protocol stack (*see, e.g.*, '818 patent at Fig. 2), where each "layer" of the stack provides services to those above it, and higher layers make use of those services through an interface. *See* '818 patent at 5:16-17 ("Virtual network interface 220 presents a virtual link layer interface to higher layers of the protocol stack."). A POSITA would therefore understand that the claimed "virtual [storage network] interface layer[s]" correspond to those depicted in Figure 2, which shows the "protocol stack and software modules of an application servers" '818 patent at 2:32-34. A POSITA further understands that a layer in a protocol stack necessarily provides an interface to the layers above it in the stack. Ex. 45, Definition of Layer, VMware-IV_000032322 ("Referring to the protocol or protocols operating at a particular level within a network architecture. Such an architecture commonly is detailed in a protocol stack . . . the bottom layer, the Physical Layer, deals with physical and mechanical aspects of the **interface** between a device and a transmission medium.") (emphasis added). Indeed, the specification does not describe any other protocol stack or interfaces to layers in protocol stacks to which these claim terms would reasonably correspond. '818 Patent at 5:16-17. IV's proposed construction should be rejected because it is contrary to Federal Circuit precedent when the parties have a claim-scope dispute. *O2 Micro*, 521 F.3d at 1361-62. VMware's proposed construction should be adopted because it is consistent with the language of the claims and intrinsic evidence.

F. "one or more input/output virtualization modules comprising computer-readable instructions operative to cause the one or more processors to" performs functions terms ('818 patent claim 17)

Due to the length of these terms, the parties' proposals are included separately in Ex. 32, instead of in a summary table herein. Each of these claim terms include the word "module" which "is a well-known nonce word that can operate as a substitute for "means." *Williamson*, 792 F.3d at 1350. The claims expand on this nonce term by reciting "input/output virtualization modules comprising computer-readable instructions operative to cause the one or more processors to" perform certain functions, but this additional language similarly "fails to impart any structural significance to the term." *Id*. at 1351. Lacking structural language, the claims should be construed as means-plus-function terms.

However, these claims reciting these means-plus-function terms are invalid as indefinite because the specification also fails to disclose a corresponding structure for performing the recited functions. *See Williamson*, 792 F.3d at 1352. Here, the specification is devoid of any algorithm in the specification or prosecution history for performing the claimed functions. Ex. 2, Snoeren Decl. ¶ 115; *See Grecia v. Samsung Elec. Am., Inc.*, 780 Fed. Appx. 912, 916 (Fed. Cir. Aug. 20, 2019) ("For a computer-implemented means-plus-function term, the corresponding structure is typically the algorithm disclosed in the specification for performing the claimed function"). Additionally, IV has failed to identify any specific structure in the specification and instead broadly cites to multiple components without tying these components to a description of how they perform the function. This is improper. *Augme Techs., Inc. v. Yahoo! Inc.*, 755 F.3d 1326, 1338 (Fed. Cir. 2014).

Since the term "input/output virtualization modules" is not a term that refers to a structure and the '818 patent fails to recite an algorithm to perform the recited function, claim 17 of the '818 patent is invalid as indefinite.

VII. CONCLUSION

For the reasons stated herein, VMware respectfully requests the Court adopt its proposed constructions for the disputed terms and phrases.

Dated: March 6, 2020

Respectfully submitted,

WINSTON & STRAWN LLP

/s/ *Katherine Vidal*
Katherine Vidal
Admitted *Pro Hac Vice*
KVidal@winston.com
Michael R. Rueckheim
Texas State Bar No. 24081129
MRuekheim@winston.com
WINSTON & STRAWN LLP
275 Middlefield Road, Suite 205
Menlo Park, CA 94025
Telephone: (650) 858-6500
Facsimile: (650) 858-6559

Thomas M. Melsheimer
Texas State Bar No. 13922550
M. Brett Johnson
Texas State Bar No. 00790975
MBJohnson@winston.com
Michael A. Bittner
MBittner@winston.com
Admitted *Pro Hac Vice*
Winston & Strawn LLP
2121 N. Pearl St., 9th Floor
Dallas, TX 75201
Telephone: (214) 453-6500
Facsimile: (214) 453-6400

Vivek V. Krishnan
Admitted *Pro Hac Vice*
VKrishnan@winston.com
DaWanna McCray
Admitted *Pro Hac Vice*
DMcCray@winston.com
Winston & Strawn LLP
35 W. Wacker Drive
Chicago, IL 60601
Telephone: (312) 558-5600
Facsimile: (312) 558-5700

William M. Logan
Texas State Bar No. 24106214
WLogan@winston.com
Winston & Strawn LLP
1111 Louisiana Street
Houston, TX 77002
Telephone: (713) 651-2766
Facsimile: (713) 651-2700

ATTORNEYS FOR DEFENDANT VMWARE, INC.

CERTIFICATE OF SERVICE

I hereby certify that, on March 6, 2020, the foregoing document was electronically filed with the Clerk of Court using the Court's CM/ECF system which will send notification of such filing to all counsel of record, including counsel of record for Plaintiffs Intellectual Ventures I LLC and Intellectual Ventures II LLC.

/s/ *Katherine Vidal*
Katherine Vidal